大数据基础知识

阿里云大数据分析师ACP认证课程

阿里云

课程目标

学习完本课程后，你将能够：

1. 了解大数据的概念、特征、发展历程、技术演进与应用场景
2. 了解开源大数据平台Hadoop的原理与其主要生态构成
3. 了解大数据分析的概念、应用场景与流程
4. 了解数据库、数据仓库的原理与区别
5. 了解大数据分析所需要的统计基础

阿里云

阿里云全球培训中心

课程目录

大数据的定义

- **大数据（Big Data）**
  - 指无法在可承受的时间范围内用常规软件工具进行捕捉、管理和处理的数据集合
  - 是需要新处理模式才能具有更强的决策力、洞察发现力和流程优化能力的海量、高增长率和多样化的信息资产。

大数据就像矿石，需要汇聚、清洗、分析、挖掘等处理才能发挥蕴含的价值



为什么会有大数据

移动互联网
带来爆炸式的数据增长

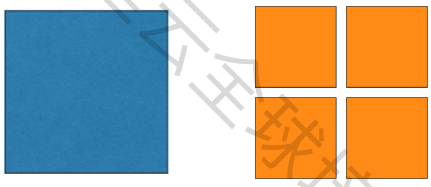
数据作为一种资产
越来越重要

存储技术和计算能力
的飞速发展

大数据的处理思路

将问题化简成一个更简单的能处理的问题

将问题拆分成多个可以简单求解的小问题

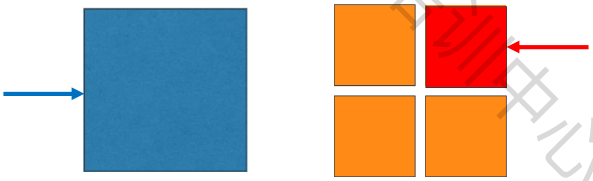


传统方式　　**大数据方式**

大数据的处理思路

将问题化简成一个更简单的能处理的问题

将问题拆分成多个可以简单求解的小问题

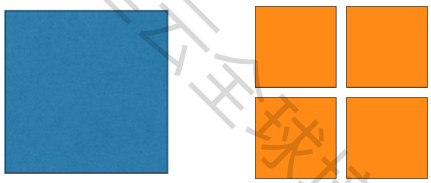


传统方式　　大数据方式

大数据的处理思路

将问题化简成一个更简单的能处理的问题

将问题拆分成多个可以简单求解的小问题

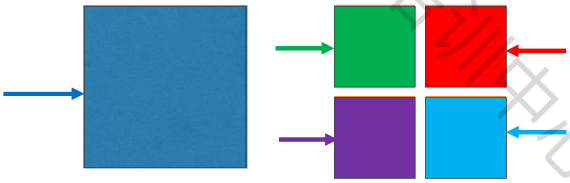


传统方式　　大数据方式

大数据的处理思路

将问题化简成一个更简单的能处理的问题

将问题拆分成多个可以简单求解的小问题



传统方式　　大数据方式

云计算与大数据

- 云计算提供存储与计算的基础设施，大数据则是运行在其上的应用。

电力系统

云计算

电器设备

大数据

大数据的产生与发展-统计学的出现

1
- **1663年**
- John Graunt利用统计学建立起对黑死病早期预警系统的理论

2
- **1865年**
- 银行家Henry Furnese用结构化的方式收集和分析了有关竞争对手的商业活动来获取竞争优势，这被认为是第一次数据分析的商业应用

3
- **1880年**
- 美国人口普查局工程师Herman Hollerith发明了打孔卡制表机，大大缩短了工作时间

大数据的产生与发展-数据大爆炸开启

4
- **2004年**
- 始于出版社经营者O'Reilly和MediaLive International之间的一场头脑风暴论坛，Web 2.0诞生，即用户生成的Web，其中大部分内容将由服务的用户提供，而不是服务提供者本身

5
- **2004年**
- Facebook这样的社交软件开始出现

6
- **2005年**
- Hadoop这个开源框架被创建出来，专门用于存储和分析大数据集。它的灵活性使它对管理非结构化数据(语音、视频、原始文本等)特别有用，我们正在越来越多地生成和收集这些数据

大数据的产生与发展-数据时代全面到来

7
- **2013年**
- 国内云计算热潮兴起，阿里云成为世界上第一家对外提供5K云计算服务能力的公司

8
- **2014年**
- Spark、Storm、Flink成为Apache开源项目，流式计算引擎逐渐投入应用

9
- **2015年**
- 国务院发布《促进大数据发展的行动纲要》，大数据上升到了国家战略层面

课程目录

大数据的特征

- **大数据（Big Data）**
  - 指无法在可承受的时间范围内用常规软件工具进行捕捉、管理和处理的数据集合
  - 是需要新处理模式才能具有更强的决策力、洞察发现力和流程优化能力的海量、高增长率和多样化的信息资产。

- **大数据的4V特征：**

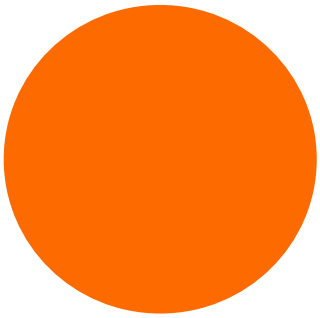
Volume 体量巨大	Variety 种类繁多	Value 价值密度低	Velocity 处理速度快

大数据的特征 – 体量巨大

- 根据IDC做出的估测，数据一直都在以每年50%的速度增长，即两年增长一倍
- 人类在最近两年产出的数据量相当于之前产生的全部数据量
- 人类生产的所有印刷材料的数据量是200PB
- 历史上全人类说过的所有的话的数据量大约是5EB(1EB=1024PB)

个人计算机硬盘的容量：TB量级

人类生产的所有印刷材料：200PB



一些大企业的数据量：EB量级

大数据的特征 – 种类繁多

- 10%为结构化数据，通常存储在数据库中
- 90%为非结构化数据，格式多种多样

科学研究	企业应用	互联网	物联网
• 基因组 • 地球与空间探测	• 交易记录 • 应用日志 • 文档、文件	• 文本 • 图像、视频 • 查询日志、点击流	• 传感器 • 监测设备

大数据的特征 – 处理速度快

- 从数据的生成到消费，时间窗口非常小，可用于生成决策的时间非常少。

- 最高峰值订单54.4万笔/秒
- 单日处理数据量达到970PB

大数据的特征 – 价值密度低

- 以视频为例，一部1小时的视频，在不间断的监控中，有用数据可能仅有几秒，价值密度低，商业价值高。


资产负债表

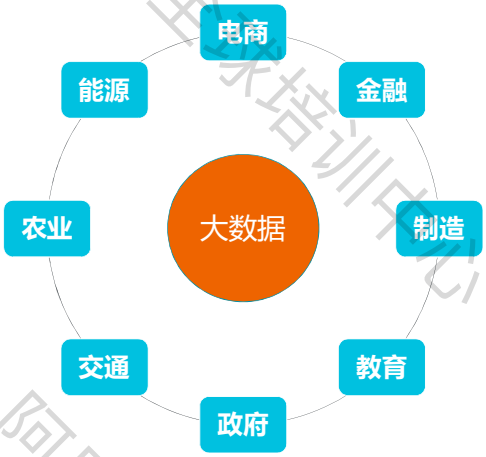
财务报表

视频

课程目录

大数据的应用领域

- 大数据在许多行业都有着广泛的应用



阿里云

大数据应用场景示例 - 城市治理

- 2016年，杭州市政府联合各阿里云等企业开始建设“城市大脑”。2017年的云栖大会上，城市大脑1.0正式发布。次年的云栖大会发布了更加强大的2.0版本。

阿里云

大数据应用场景示例 - 医疗

- 大数据预测早期癌症
  - 通过对血常规、血生化和尿常规数据进行分析预测，为每个体检者预测出一个患癌症的风险几率。
  - 然后将预测结果与癌症患者已有指标进行对比，采用**大数据**标准评估和**人工智能**模拟技术，动态评估预测体检者的早期癌症风险。

预测所需的常规血尿数据

血常规	生化全套	尿常规
红细胞	白蛋白	潜血
白细胞	谷丙转氨酶	胆红素
红细胞压积	谷草转氨酶	维生素C
淋巴细胞绝对值	γ-谷氨酰转肽酶	葡萄糖
淋巴细胞百分比	总胆红素	酮体
中性粒细胞绝对值	尿素氮	比重
中性粒细胞百分比	肌酐	酸碱度/PH值
血小板比积	尿酸	蛋白质
单核细胞绝对值	葡萄糖	尿胆素原
单核细胞百分比	总胆固醇	亚硝酸盐

备注：上表只是部分数据

课程目录

Hadoop概述

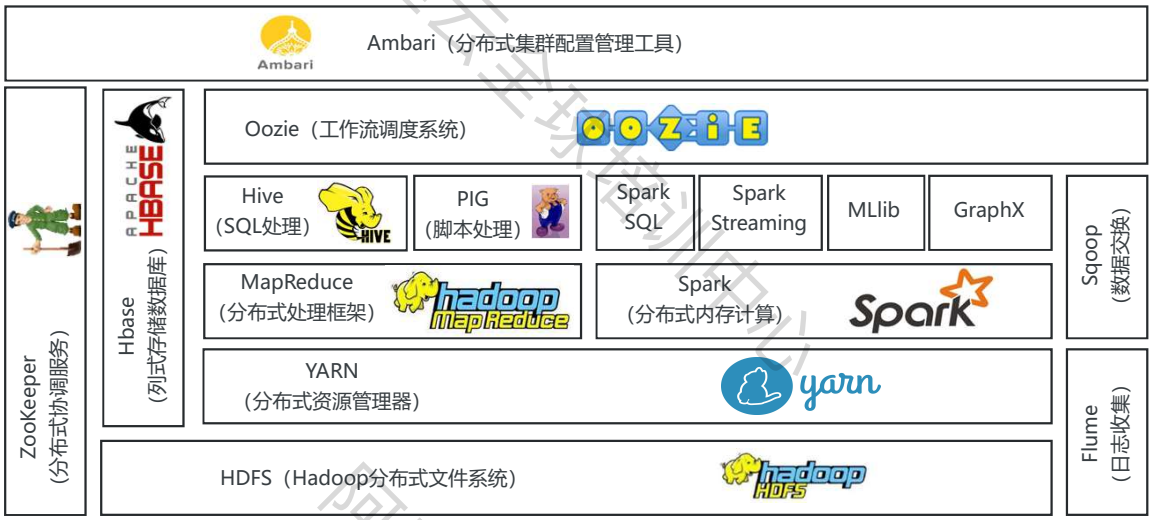
- Hadoop是一个在计算机硬件的集群上存储数据、运行应用程序的开源的软件框架。
- 解决大数据运算的框架方案
- 大规模的存储/计算
- 对于大型的Job处理速度非常快
- 多种多样的处理引擎,如:Tez、Spark、Storm等

	传统的关系型数据库	Hadoop
模式	着重数据写入	着重数据读出
速度	读取速度更快	写的速度很快
数据治理	标准的结构化的数据	可以不是结构化标准的
处理	有限的，没有数据处理	数据处理
数据类型	结构化	结构化 半结构化 非结构化
更适用于	交互式的OLAP分析 复杂的ACID事务 数据操作	数据挖掘 处理非结构化数据 大量的数据存储/处理

Hadoop的特点

- **高可靠性:**
  - Hadoop按位存储和处理数据的能力值得人们信赖
- **高扩展性:**
  - Hadoop在计算机集群间分配数据并完成计算，可以方便地扩展到数以千计的节点中
- **高效性:**
  - Hadoop能够在节点之间动态地移动数据，因此处理速度非常快
- **高容错性:**
  - Hadoop能够自动保存数据的多个副本，并且能够自动将失败的任务重新分配
- **低成本:**
  - 与一体机、商用数据仓库以及数据集市相比，Hadoop是开源的，项目的软件成本因此会大大降低

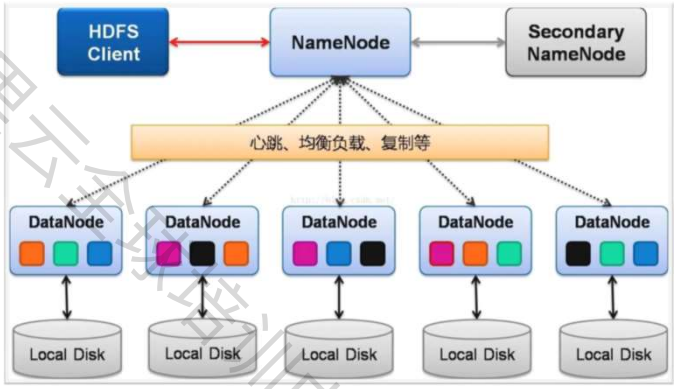
Hadoop生态概览



HDFS文件系统

HDFS是Hadoop Distributed File System的简称，是Hadoop抽象文件系统的一种实现。

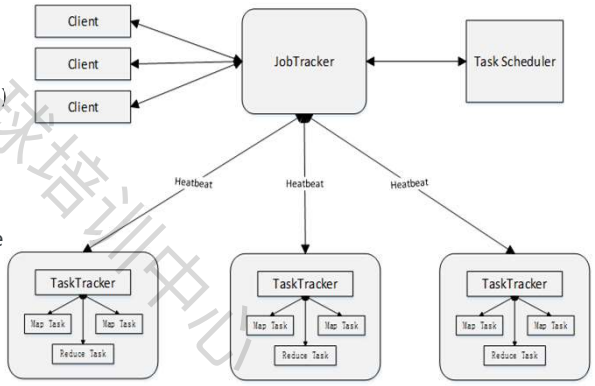
- 适合海量数据存储
- 适合大数据量批处理
- 一次性写入，多次读取
- 保证数据一致性
- 通过多副本提高可靠性



MapReduce

Apache MapReduce

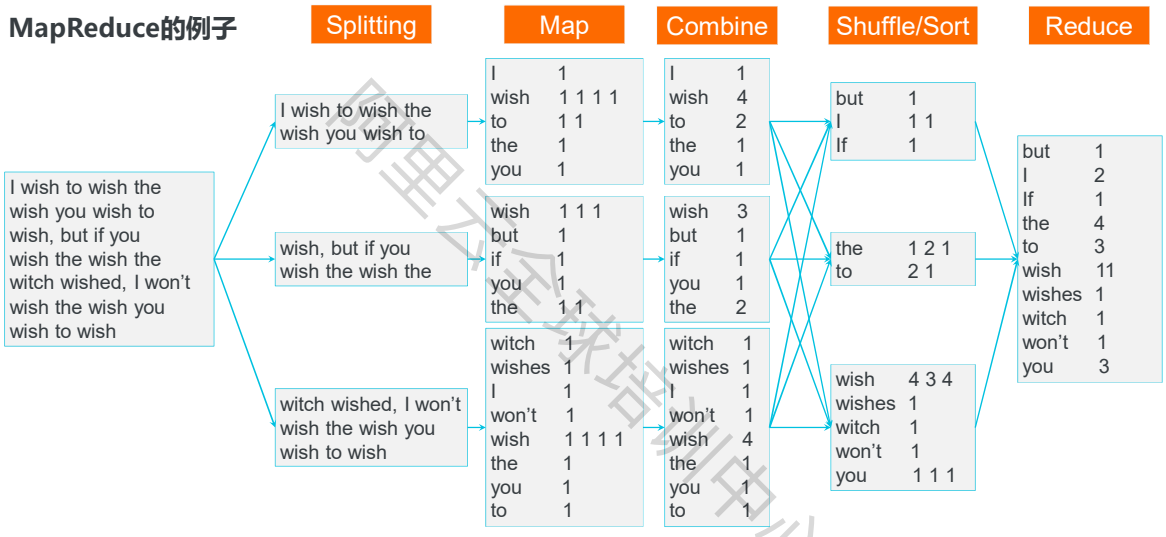
- 是从 Google MapReduce 派生而来的
- MapReduce是一种编程模型，用于大规模数据集（大于1TB）的并行运算
- 适合大规模数据处理
- 每个node处理数据后都将会存储在这个node上
- 每个MapReduce job都是由两个阶段组成的：Map/Reduce



阿里云

分布式离线计算框架 - MapReduce

MapReduce的例子

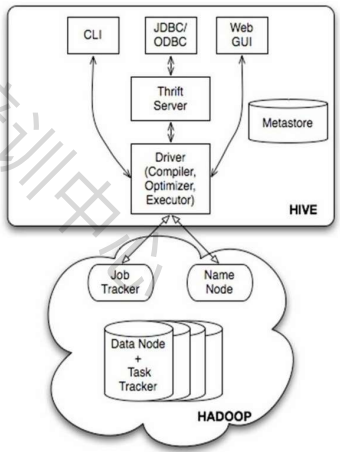


阿里云

Hive

Hive是基于Hadoop的一个数据仓库工具，用来进行数据提取、转化、加载，这是一种可以存储、查询和分析存储在Hadoop中的大规模数据的机制。

- 操作接口采用类SQL语法，学习成本低
- 避免了去写MapReduce，开发效率高
- 适用于海量结构化数据离线分析
- WEB访问日志分析（典型应用）
- 可用于构建离线数据仓库



HBase

Hbase是一个高可靠、高性能、面向列、可伸缩的分布式数据库，是谷歌BigTable的开源实现，主要用来存储非结构化和半结构化的松散数据。

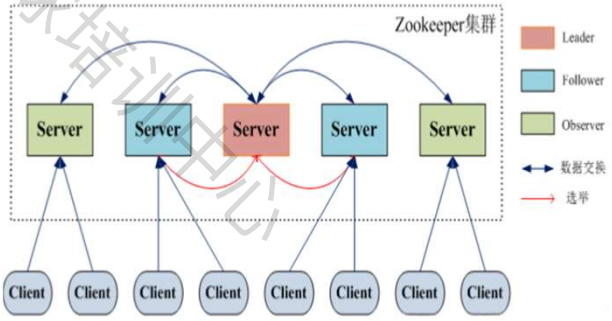
- 海量存储，列式存储，高并发
- 极易扩展，可以横向添加RegionSever的机器，进行水平扩展
- 可以支撑高并发KV查询场景
- 可以支撑实时或批量数据更新

	HBase	RDBMS
数据类型	只有字符串	丰富的数据类型
数据操作	简单的增删改查	各种各样的函数，表连接
存储模式	基于列存储	基于表格结构和行存储
数据保护	更新后旧版本仍然会保留	替换
可伸缩性	轻易的进行增加节点，可扩展性高	需要中间层

ZooKeeper

ZooKeeper是一个分布式服务框架，是Apache Hadoop 的一个子项目，主要是用来解决分布式应用中经常遇到的一些数据管理问题。

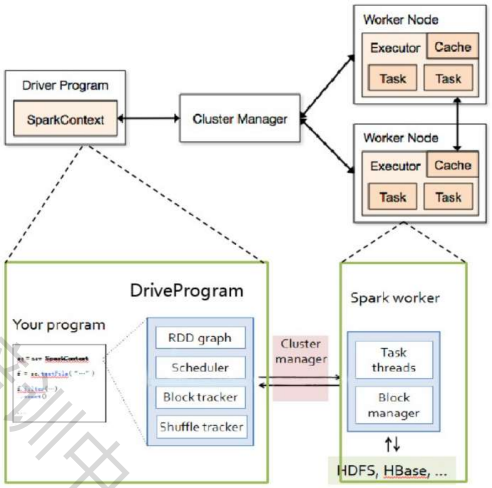
- 在Hadoop中，ZooKeeper主要用于实现HA(High Availability），包括HDFS的NamaNode和YARN的ResourceManager的HA
- 在YARN中，ZooKeeper用来存储应用的运行状态
- 还可用于负载均衡、数据的发布和订阅等



Spark

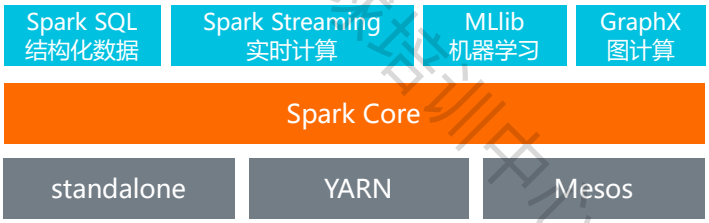
Spark

- 是UC Berkeley AMP lab (加州大学伯克利分校AMP实验室)所开源的类Hadoop MapReduce的通用并行框架
- 专门用于大数据量下的迭代式计算。



Spark

- 伯克利将 Spark 的整个生态系统分成为伯克利数据分析栈（BDAS），在核心框架 Spark 的基础上，主要提供四个范畴的计算框架：



- Spark SQL：提供了类 SQL 的查询,返回 Spark-DataFrame 的数据结构
- Mllib： 提供机器学习的各种模型和调优
- Spark Streaming：流式计算,主要用于处理线上实时时序数据
- GraphX：提供基于图的算法

课程目录

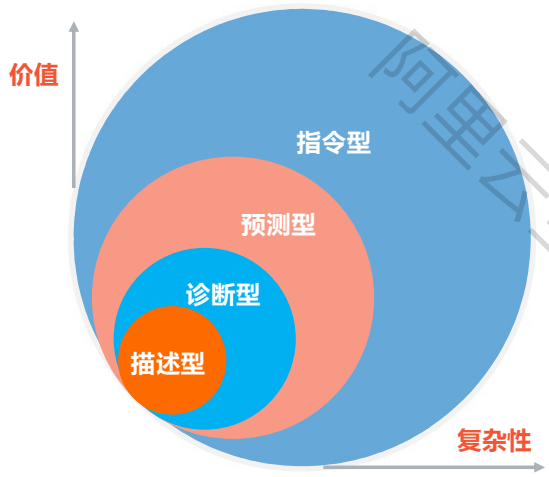
大数据分析的概念

- **数据分析**是基于**商业等目的**，有目的的进行收集、整理、加工和分析数据，**提炼有价值信息**的过程。
- **大数据分析**针对的是海量的多样化的数据集合。

需求分析 明确目标 → 数据收集 加工处理 → 数据分析 数据展现 → 分析报告 提炼价值

阿里云

大数据分析方法分类



- **描述型分析**：发生了什么？
  - 广泛的，精确的实时数据
  - 有效的可视化
- **诊断型分析**：为什么会发生？
  - 能够钻取数据的核心
  - 能够对混乱的信息进行分离
- **预测型分析**：可能发生什么？
  - 使用算法确保历史模型能够用户预测特定的结果
  - 使用算法和技术确保自动生成决定
- **指令型分析**：下步怎么做？
  - 依据测试结果来选定最佳的行为和策略
  - 应用先进的分析技术帮助做出决策

阿里云

课程目录

阿里云

大数据分析的应用场景 – 概述

- 大数据分析，在商业、制造业、媒体等领域有广泛的应用

商业	• 零售商、银行、制造商、电信供应商和保险公司等都在利用数据挖掘技术。
制造业	• 制造业存在大量的设备，大量的数据构建制造业的大数据， 作为设备诊断和健康管理分析工具的输入项。
媒体	• 媒体的大数据应用，针对不同用户调整消息(广告)和内容(文章)。

阿里云

大数据分析的应用场景举例 – 关联分析

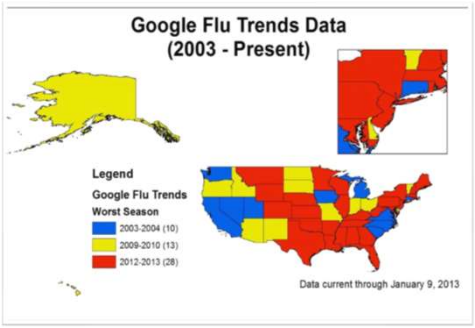
- **啤酒与尿布**
  - 啤酒与尿布的故事可以说是营销界的经典段子。这个故事产生于20世纪90年代的美国沃尔玛超市中，沃尔玛的超市管理人员在分析销售数据时发现了一个令人难以理解的现象: 在某些特定的情况下，啤酒与尿布两件看上去毫无关系的商品会经常出现在同一个购物篮中，而且，啤酒与尿布在周末的时候销量明显会高于平时。这种独特的销售现象引起了管理人员的注意。他们经过后续调查发现，这种现象出现在年轻的父亲身上。

- 为什么“啤酒与尿布”的故事会产生在沃尔玛的卖场中?
- 原因来自于两个方面:
  - 沃尔玛先进的计算机技术
  - 沃尔玛运用了大数据分析方法

大数据分析的应用场景举例 – 趋势预测

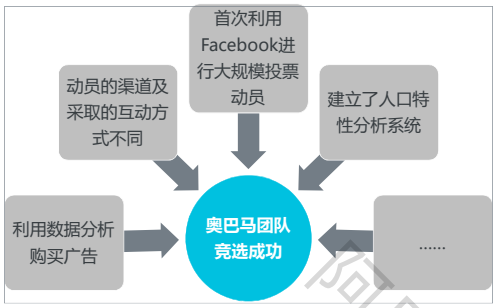
- **谷歌流感趋势**
  - 谷歌流感趋势是谷歌2008年推出，用于预警流感的即时网络服务。其发明者是谷歌公司的两名软件工程师杰瑞米·金斯伯格和马特·莫赫布。他们一致认为:”谷歌搜索显示的数据分布模式非常有价值“。

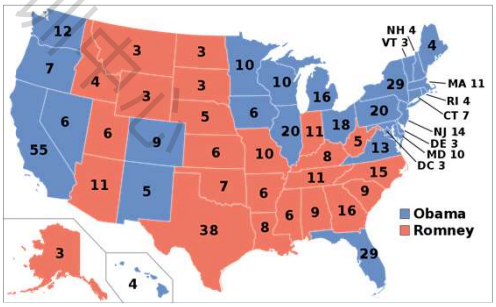
- 谷歌在美国的九个地区就这一观点做了测试，并且发现，它比联邦疾病控制和预防中心提前了7~14天准确预测了流感的爆发。



大数据分析的应用场景举例 – 决策支持

- **美国总统大选**
  - 美国总统奥巴马成功击败对手罗姆尼，再次赢得美国总统选举的当天，《时代》杂志撰写了一篇文章，描述了奥巴马总统获胜背后的秘密数据挖掘。 以竞选工作组发言人本拉波特的话来形容:“奥巴马团队拥有‘核代码’，数据是能够击败罗姆尼的最根本优势!





课程目录

大数据分析流程总览

课程目录

大数据采集技术 – 大数据采集的方法

基于浏览器	无线客户端	物联网	源业务系统	第三方数据
网络爬虫/API	采集SDK/埋点	测量值转化数字信号	数据同步	合作方提供
页面浏览日志	**页面浏览事件**	**如人工智能驾驶**	**结构化数据**	**如政府公布宏观数据**
页面交互日志	**控件点击事件**	**（温湿度、障碍物）**	**客户、交易等**	**公安系统的身份核验**

阿里云

大数据预处理技术 – 数据清洗

数据清洗：

- 针对原始数据，对出现的噪声进行修复、平滑或者剔除
- 包括异常值、缺失值、重复记录、错误记录等
- 同时过滤掉不用的数据，包括某些行或某些列

噪声数据处理：

- 异常值：箱线图、Tukey's Test等
  - 删除、当做缺失值、忽略
  - 分箱发：箱均匀、箱中位数或箱边界、平滑数据
- 缺失值
  - 统计值填充：均值、众数、中位数
  - 固定值填充：填充指定值
  - 最接近记录值填充：与该样本最接近的相同字段值
  - 模型拟合填充：填充回归或其他模型预测值
  - 插值填充：建立插值函数，如拉格朗日插值法、牛顿插值法等

阿里云

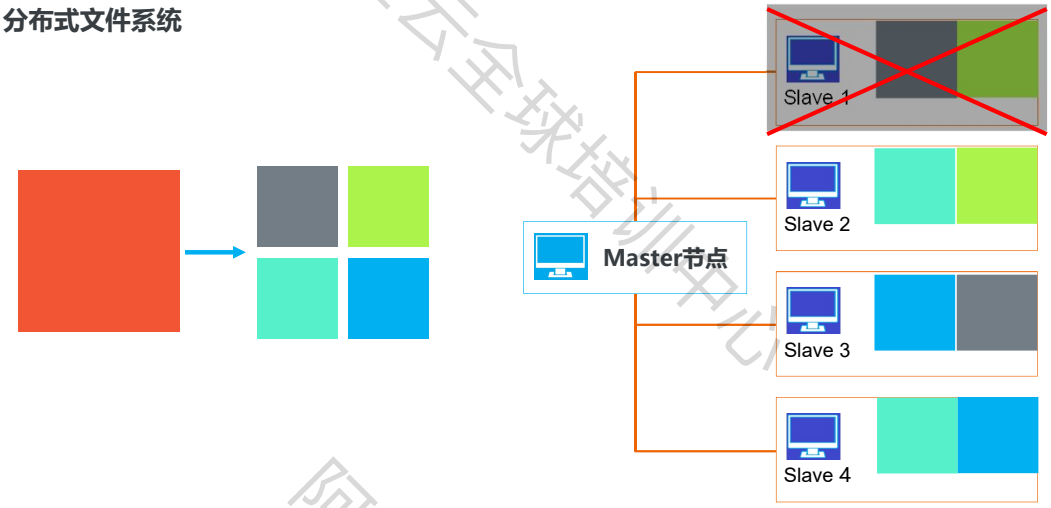
课程目录

大数据的存储

- 单硬盘存储：使用一块硬盘存储数据
- 磁盘阵列（RAID）：在单机上使用多块硬盘均匀储存数据
- 分布式存储：使用多台机器基于网络连接储存数据

大数据的存储技术

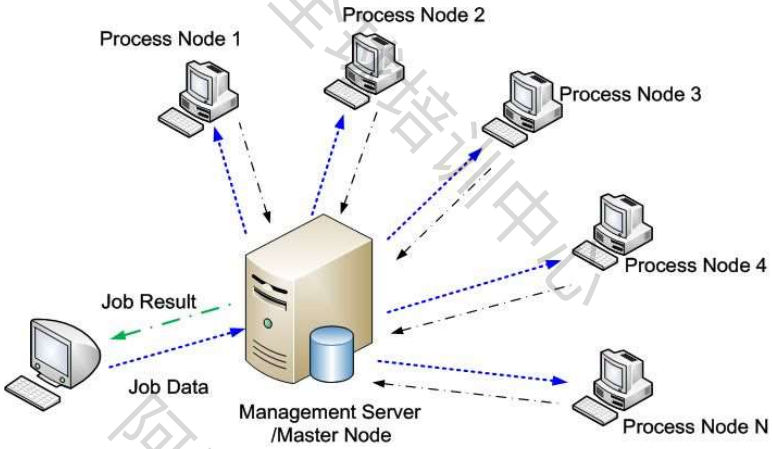
- 分布式文件系统



课程目录

大数据的计算

- 大数据时代的计算模式，从单机计算进入了集群计算



大数据的计算技术

计算技术	计算模式	特点	主要应用场景
批量计算	数据批量装载，外界发起计算请求，一次计算一批数据	批量 高延时 主动触发计算	实时性要求不高的离线计算场景。 如：离线分析、报表、数据挖掘等
流式计算	计算作业运行等待，数据流到计算作业触发计算，一次计算一条数据	持续 低延时 事件触发计算	实时性要求高的实时计算场景。 如：实时推荐、业务监控等

大数据计算技术 – 离线批处理

传统关系型数据库

- SQL
- 脚本语言（shell、tcl等）
- 编程语言（java、c等）

Hadoop开源平台

- MapReduce
- Hive SQL
- Spark

阿里云 MaxCompute +DataWorks

- MaxCompute SQL
- MapReduce
- Graph

大数据计算技术 – 实时流处理框架平台

- **Twitter开源的分布式实时大数据处理框架**
  - 框架简单，学习成本低
  - 毫秒级延迟，实时性好
  - 健壮、稳定
  - 编程成本高
  - 逻辑与批处理完全不同，无法公用代码
  - Debug比较复杂

- **Spark核心API的扩展，实现高吞吐、可容错的实时流处理**
  - 编程语言丰富、编程简单
  - 框架封装层级高，封装性好
  - 可以共用批处理逻辑
  - 微批处理，时间延迟大
  - 稳定性相对较差
  - 机器性能消耗大

- **Flink是一种可以处理批处理任务的流处理框架**
  - Flink流处理为先的方法可提供低延迟、高吞吐、近乎逐项处理的能力
  - 可通过多种方式对工作进行分析进而优化任务
  - 提供了基于Web的调度视图

课程目录

大数据可视化

- **数据可视化**
  - 利用计算机图形学和图像处理技术，将数据转换为图形或者图像在屏幕上显示出来进行交互处理的理论方法和技术
  - 数据可视化主要旨在借助于图形化手段，清晰有效地传达与沟通信息。

- 数据可视化随着平台的拓展、应用领域的增加，表现形式的不断变化
- 从原始的BI统计图表，到不断增加的诸如实时动态效果、地理信息、用户交互等等
- 数据可视化的概念边界不断扩大

大数据BI

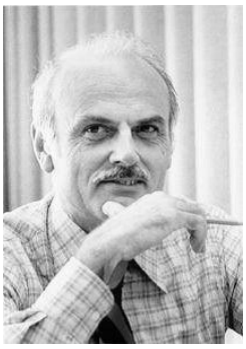
商业智能BI（Business Intelligence）是用现代数据仓库技术、线上分析处理技术、数据挖掘和数据展现技术进行数据分析以实现商业价值。

现代BI工具：

- 阿里云Quick BI: 丰富图表、海量数据实时在线分析、集成方便、成本低、使用简单。
- Tableau: 图表制作能力强、探索分析；但大数据支持不足、硬件要求高、价格高。
- QlikView: 开发使用简单、多数据源；但硬件要求高、界面美观性一般、价格较高。

课程目录

关系型数据库的出现



Edgar Frank Codd (1923-2003)

- 1970年IBM的埃德加·科德（Edgar Codd，1981年图灵奖获得者）发表了跨时代的著名论文，开启了关系数据库的时代，成为了大多数数据库系统的基础
- 在计算机的辅助下，人类处理信息地速度空前地加快了

数据库的相关概念：数据库系统的构成、数据库设计三范式、关系模型

数据库系统基本知识

- **Hbase是分布式数据库，那么到底什么是数据库?**
- 传统的数据库系统中最常见的是关系型数据库（Relational Database）
- 常见的数据库引擎：

ORACLE 甲骨文 MySQL Microsoft SQL Server IBM DB2 PostgreSQL

- 数据库系统的构成
  - 支持数据库运行的软、硬件环境
  - 数据库
  - 数据库管理系统
  - 用户
- 数据库设计的三范式
  - 1NF：原子性
  - 2NF：唯一性
  - 3NF：独立性

关系型数据库

- 关系模型是目前应用最广的数据模型。在关系模型中，用一个二维表来表示数据间的关系。

课程编号	课程名称	学时	学分	课程性质
B21112003	体育	32	1	通识必修
B08711002	英语	72	4.5	通识必修
B04811124	计算机导论	64	4	学科基础
B20112021	概率论与数理统计	56	3.5	学科基础
B04811037	C程序设计	64	4	学科基础
B04211026	计算机网络原理	64	4	专业必修
B04422073	人工智能基础	32	2	专业选修

关系模型的基本概念

- **关系**：就是一张二维表，每个关系有一个关系名
- **元组**：二维表中水平方向的行称为元组
- **属性**：二维表中垂直方向的列称为属性（Attribute）。
- **域**：域（Domain）是指属性的取值范围。
- **主键**：主键（Primary Key）是指表中的某个属性或某些属性的集合能唯一地确定一个元组。
- **外键**：外键（Foreign Key）是一个表中的一个属性或属性组，而它们在其它表中却作为主键存在，即一个表中外键被认为是另一个表中的主键
- **五大约束**：主关键字约束，外关键字约束，唯一性约束，检查约束，缺省约束

关系模型

课程号	课程名称	学时	学分	课程性质
A002	体育	32	1	通识必修
A003	英语	72	4.5	通识必修
B001	计算机导论	64	4	学科基础
B002	概率论与数理统计	56	3.5	学科基础
B048	C程序设计	64	4	学科基础
B042	计算机网络原理	64	4	专业必修
B044	人工智能基础	32	2	专业选修

元组

外键

主键

属性

学 号	课程号	平时成绩	考试成绩
0100001	A002	90	90
0100001	B001	95	92
0100002	A002	85	90
0100003	B022	100	98

- **数据库与数据仓库中存储的数据如何进行分析?**

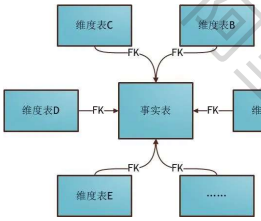
课程目录

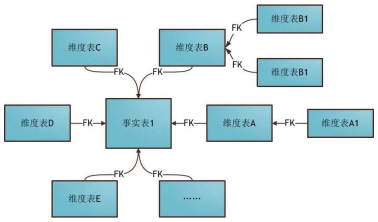
数据仓库

- **数据仓库**：
  - 一个面向主题的、集成的、随着时间变化的、用于管理决策支持的不可更新的数据集合
- **数据仓库的特点**：
  - 面向主题：数据仓库都是基于某个明确主题，仅需要与该主题相关的数据
  - 集成：从不同的数据源采集数据到同一个数据源，一致的命名约定、格式、编码结构
  - 随时间变化：关键数据隐式或显式地基于时间变化，可以研究趋势和变化
  - 不可改变：只读，定期刷新。数据仓库的数据反映的是一段相当长的时间内历史数据的内容
- **数据仓库的相关概念**：
  - 数据仓库模型
  - ETL
  - OLTP/OLAP

数据仓库模型

- 在多维分析解决方案中，根据事实表和维度表的关系，可将常见的模型分为星型模型和雪花型模型。
- 在设计逻辑型数据的模型的时候，就应考虑数据是按照星型模型还是雪花型模型进行组织。





- **星形模式**：最常用的维度建模方式，由一个事实表和一组维度表组成
  - 维度表只和事实表关联，维度表之间无关联关系
  - 每个维度表的主键均为单列，且该键放置在事实表中，作为两表关联的外键
  - 以事实表为核心，维度表围绕核心呈星形分布
- **雪花模式**：对星形模式的扩展，每个维表可继续向外连接多个子维度表。
  - 雪花模型相当于将星形模式的大维度表拆分成小维度表，满足了规范化设计。
  - 雪花模型仅是一种理论上的模型，在实际应用中很少见，因为这样做会导致开发难度增大，而在数据仓库中并不太注重数据冗余问题

ETL流程

ETL是构建数据仓库的重要一环，用户从数据源**抽取（E**xtract）出所需的数据，经过数据**清洗（T**ransform），最终按照预先定义好的数据仓库模型，将数据**加载（L**oad）到数据仓库中去。

抽取
- 关系型数据库
- Hadoop
- 结构化文件
- 非结化文件
- 其他（可扩展）

清洗
- 定义清洗函数
- 文件处理

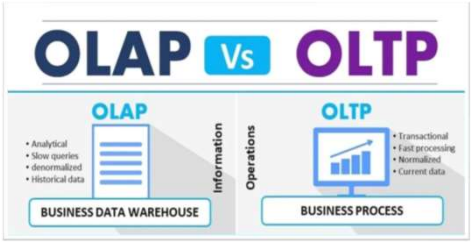
加载
- 关系型数据库
- Hadoop

转换
- 单表处理
- 多表关联
- 其他

规则
- 检查规则
- 实时告警
- 中断式告警

OLTP与OLAP

- **OLTP：OLTP是一个在线交易处理系统。**
  - OLTP系统的主要是记录事务时的当前更新，插入和删除。
  - OLTP查询更简单，更短，因此需要更少的处理时间，也需要更少的空间。
  - OLTP数据库经常更新
  - OLTP数据库具有规范化表（3NF）
  - OLTP中的数据处理可能会在中间失败，这可能会影响数据完整性。



- **OLAP：OLAP是一个在线分析处理系统。**
  - OLAP数据库存储OLTP输入的历史数据
  - OLAP允许用户执行复杂的查询以提取多维数据
  - 在OLTP中，即使数据处理在中间失败，也不会损害数据完整性
  - OLAP执行的任务很长，需要花费更多的时间进行处理，并且需要很大的空间
  - 与OLTP相比，OLAP中的事务较少。甚至OLAP数据库中的表也可能不规范

数据库与数据仓库的区别

数据仓库与数据库的主要区别在于：

- 数据库是面向事务的设计，数据仓库是面向主题设计的
- 数据库一般存储在线交易数据，数据仓库存储的一般是历史数据
- 数据库尽量避免冗余，数据仓库有意引入冗余
- 数据库为捕获数据而设计，数据仓库为分析数据而设计

特性	数据库	数据仓库
数据	当前数据	历史数据
面向	业务操作	数据分析
存取	读写操作	多为只读
使用频率	高	较低
数据访问量	少	多
要求的响应时间	较短	可以很长
关注	数据输入	信息输出

课程目录

大数据分析所需要的统计知识框架

- 基本概念：总体/样本、参数/统计量、 变量、频率/概率
- 数据的概括性度量：
  - 集中趋势：众数、中位数、平均数、分位数
  - 离散趋势：方差/标准差、极差、四分位差、变异系数
- 常见的概率分布：
  - 离散型分布：二项分布、泊松分布
  - 连续型分布：正态分布、均匀分布、指数分布
- 假设检验：p-value、两类错误、置信区间

基本概念

- 总体与样本：**总体**是考察对象的全体；**样本**是观测或调查的一部分个体
- 随机变量：设 Ω ={ω}为某随机现象的样本空间，称定义在Ω上的实值函数X=X(ω)为**随机变量**
- 随机变量的类型：**离散型**与**连续型**
  - 若随机变量可能取值的个数为有限个或可列个，则称X为**离散随机变量**
  - 若随机变量X的可能取值充满某个区间[a, b]，则称X为**连续随机变量**
- 频率与概率：**概率**是某一事件发生的可能性，**频率**是通过实验得出的概率的估计值
- 概率的基本性质：
  - 非负性：对任意事件，概率总是分布在[0,1]的区间内
  - 正则性：事件不同情况发生概率的总和等于1
  - 可列可加性：互斥事件的概率满足加法式
- 概率分布：用于表述随机变量取值的概率规律
- 概率分布函数：设X为一个随机变量，对任意实数x，称 $F(x) = P(X \leq x)$为X的分布函数

数据的概括性度量

通过计算定量数据的常见统计指标，了解数据的状况。通常从集中趋势、离散趋势两个角度进行。

- **集中趋势**：用来描述一组数据向某一中心值靠拢集中的程度。主要包括以下测量值
  - 平均数：算数平均数、几何平均数、调和平局数，加权众数：数据中出现最多的观测值
  - 中位数：是将一组数据按一定顺序排列后，处于中间位置上的变量值
  - 分位数：衡量数据位置的测定指标，主要包括上四分位数、下四分位数等
- **离散趋势**：反应变量远离集中趋势测量值的程度。主要包括以下测量值
  - 极差：一组数据最大值和最小值的差，容易受极端值影响
  - 四分位差：上四分位值与下四分位值的差，反应50%数据的离散程度
  - 方差与标准差：反应一个数据与本组数据平均值相比相差的数值
  - 变异系数：原始数据标准差与原始数据平均数的比，常用来比较两组不同数据的的离散程度

数据的概括性度量

学生排名	学生成绩Y	
1	100	
2	100	
3	98	
4	**97**	上4分位数
5	95	
6	91	
7	91	
8	**88**	中位数
9	87	
10	84	
11	75	
12	**69**	下4分位数
13	62	
14	59	
15	54	

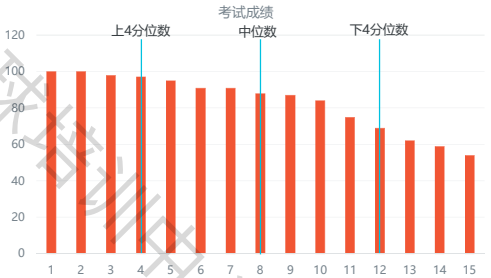
样本观测值的总和除以样本数量

平均数 $\bar{Y} = \sum_{i=1}^{15} Y_i /15$

=(100+100+98+97+…+59+54)/15

=1250/15

=83.33



数据的概括性度量

学生排名	学生成绩Y
1	100
2	100
3	98
4	**97**
5	95
6	91
7	91
8	**88**
9	87
10	84
11	75
12	**69**
13	62
14	59
15	54

上4分位数

中位数

下4分位数

平均数$\bar{Y}$ =83.33

每个样本与平均值之差绝对值的和

标准差$\sigma = \sum_{i=1}^{15}(|\bar{Y} - Y_i|)$

=(|100-83.33|+|100-83.33|+|98-83.33|

+...+|59-83.33|+|54-83.33|)/15

= 15.65

标准差的平方 → 方差σ^2 = 244.95

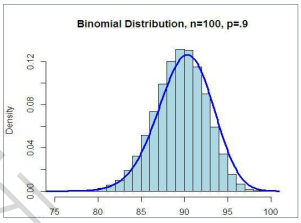
最大值与最小值的差 → 极差$R = \max(Y) - \min(Y) = 100 - 54 = 46$

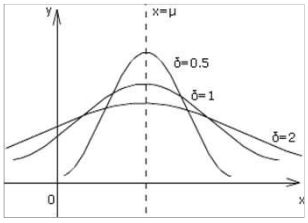
上下四分位数的差 → 四分位差$Q = Q_3 - Q_1$=97-69=28

标准差除以均值 → 变异系数 = $\sigma/\bar{Y}$ = 15.65/83.33= 0.1878

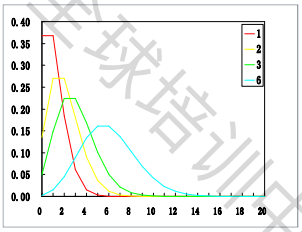
常见的概率分布

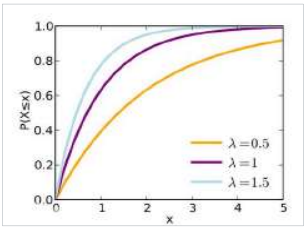
- 常见的离散型分布：
  - 二项分布：X ~ *b*(*n*, *p*)
  - 泊松分布： X ~ P(λ)





- 常见的连续型分布：
  - 正态分布： X ~ N(μ, σ2)
  - 均匀分布： X ~ U(a, b)
  - 指数分布：X ~ Exp(λ)





正态分布的概念与性质

- 正态分布
  - 正态分布是概率统计中最重要的一种分布，很多随机变量都可以用正态分布来描述，例如，人的身高、年降雨量、考试成绩。
    记随机变量X服从正态分布$N(\mu, \sigma^2)$,则其分布列为：
  - 正态分布的期望：E（X）= μ
  - 正态分布的方差：Var(X)=σ^2

$$F(x)=\frac{1}{\sqrt{2\pi}\sigma}e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

- 标准正态分布
  - 当μ=0，σ^2=1，时，称X服从标准分布

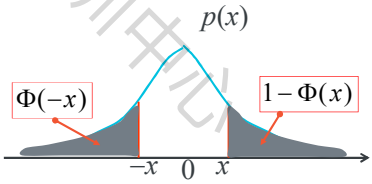
$$Y=\frac{X-\mu}{\sigma}$$ 则 Y ~ N(0, 1)

- 正态分布的3σ原则
  - 设 $X \sim N(\mu, \sigma^2)$，则：

    $P(|X-\mu|<\sigma)=0.6828$

    $P(|X-\mu|<2\sigma)=0.9545$

    $P(|X-\mu|<3\sigma)=0.9973$



阿里云

假设检验

- 假设检验问题
  - 有一类重要的统计推断问题: 根据样本的信息检验关于总体的某个假设是否正确。这类问题称作**假设检验**问题。

  - 这里有两个假设，一个叫做**原假设**，另一个叫做**备择假设。**
  - 这个检验过程从假设**原假设**是**正确**的开始。
  - **P-value**是一种在原假设为真的前提下出现观察样本以及更极端情况的概率，也叫**显著性水平**
  - **置信区间**为在某显著性水平下，样本总体某个参数的可接受范围
  - 在任何的检验中，有**两类错误。**第一类是原假设正确却拒绝它，第二类错误是当原假设不正确时却未能拒绝。

阿里云

课程目标

学习完本课程后，你将能够：

1.了解SQL基础知识

2. 掌握SQL简单应用

课程目录

1. **SQL简介**
2. SQL基础操作
3. SQL高级应用

SQL是什么

- 结构化查询语言(Structured Query Language)，简称SQL，是数据库编程的核心语言，是数据分析的必备工具

一体化

SQL 集数据定义 DDL、数据操纵 DML和数据控制 DCL 于一体，可以完成数据库中的全部工作

使用灵活

有两种使用方式，即可以直接以命令方式交互使用；也可以嵌入使用，嵌入到C、JAVA等主语言中使用

非过程化

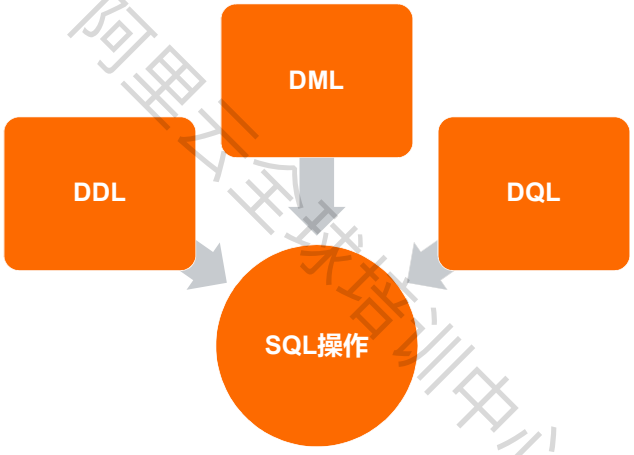
只提操作要求，不必描述操作步骤，也不需要导航。使用时只需要告诉计算机做什么，而不需要告诉它怎么做

简单易用

简洁、简单，易学好用，在 ANSI 标准中，只包含了 94 个英文单词，核心功能只用 6 个动词，语法接近英语口语

SQL分类

- SQL语言有以下三大类：数据查询语言DQL，数据操纵语言DML，数据定义语言DDL。



SQL与数据分析

- 常见应用方式：
  - 关系型数据库管理系统RDBMS，如MySQL
  - Hadoop大数据平台，如Hive
  - 阿里云大数据平台MaxCompute，相对于Hadoop平台，成本低、更易用、功能更强大、更安全

小规模一般统计分析，通过客户端工具将数据导入到MySQL，使用SQL语言及内置函数实现数据清洗、加工、统计

海量数据离线分析,通过Sqoop等工具将数据库数据导入Hive，通过Hive SQL完成数据清洗、加工、统计

海量数据离线分析,通过数据集成、客户端等工具将数据同步到MaxCompute。通过DataWorks开发MaxCompute SQL完成数据清洗、加工、统计

SQL与RDBMS

- 关系数据库管理系统（RDBMS）全称Relational Database Management System，是为管理数据库而设计的软件系统，一般具有存储、截取、安全保障、备份等基础功能
  - 数据模型是数据库系统的核心和基础；借助于集合代数等数学概念和方法来处理数据，包括选择、投影、连接、并、交、差、除、增删查改等实现对数据的存储和查询；
  - 一个关系型数据库就是由二维表及其之间的联系组成的一个数据组织，通过SQL实现数据操作；
  - 关系数据库使用主外键机制维护维护关系数据库的完整性；
  - 相对于以往的文件系统，数据库系统使用更方便；相对于其他模型数据库关系模型更贴近逻辑世界。

课程目录

阿里云

DDL综述

- DDL(Data Definition Language): 数据定义语言，用于创建、修改、删除数据库中的对象：
  - CREATE: 创建
  - ALTER: 修改
  - DROP: 删除
  - TRUNCATE 清空操作
  - COMMENT 备注处理
  - RENAME：更名操作

阿里云

DDL建库

- 使用create database或create schema命令可以创建数据库。其语法结构如下：

```
create {database|schema}[if not exists]databasename
[default]character set charset_name
|[default]collate collation_name;
```

【例】创建一个名称为mysqltest的数据库。

```
create  database if not exists mysqltest;
```


DDL建库

- 在创建数据库时，数据库命名有以下几项规则（MySQL为例）：
  - 不能与其他数据库重名，否则将发生错误。
  - 名称可以由任意字母、阿拉伯数字、下划线（_）和“$”组成，可以使用上述的任意字符开头，但不能使用单独的数字，否则会造成它与数值相混淆。
  - 名称最长可为64个字符，而别名最多可长达256个字符
  - 不能使用关键字作为数据库名、表名。

说明：在默认情况下，Windows下数据库名、表名的大小写是不敏感的，而在Linux下数据库名、表名的大小写是敏感的。如果为了便于数据库在平台间进行移植，可以采用小写来定义数据库名和表名。

DDL建表

- 创建表的语法结构：

```
create [temporary]table[if not exists]table_name
[([column_definition], ...|[index_definition])]
[table_option][select_statement];
```

- column_definition：字段的定义，包括指定字段名、数据类型、是否允许空值，指定默认值、主键约束、唯一性约束、注释字段名、是否为外键，以及字段类型的属性等。字段定义具体格式如下：

```
col_name  type  [not null | null] [default default_value]
[auto_increment] [unique [key] | [primary] key]
[comment 'string'] [reference_definition]
```


DDL建表

- 创建表示例：

```
create table `t_test_tmp` (
  `id` bigint(20) not null auto_increment,
  `app_id` varchar(128) not null comment '公众号app_id',
   primary key (`id`),
  unique key `t_test_tmp` (`app_id`)
) comment= '公众号数据登记表' ;
```

DDL修改数据表

- 修改表的语法结构：

```
alter [ignore] table tbl_name
alter_specification [, alter_specification] ...
alter_specification:
add [column] column_definition [first | after col_name ] //添加字段
|alter [column]col_name{set default literal|drop default}//修改字段默认值
|change [column] old_col_name column_definition          //重命名字段
[first|after col_name]
|modify [column]column_definition[first|aftercol_name]  //修改字段数据类型
|drop [column] col_name                                  //删除列
|rename [TO] new_tbl_name                                //对表重命名
|order by col_name                                      //按字段排序
|convert TO character set charset_name[collate collation_name]//将字符集转换为二进制
|[default] character set charset_name [collate collation_name]//修改表的默认字符集
```


DDL修改数据表

- 修改表用于修改原有表的结构信息：
  - 增加字段。在创建表时，表中的字段就已经定义完成；如果要增加新的字段，可以通过alter table语句进行增加
  - 修改表名。表名可以在一个数据库中唯一的确定一张表。数据库系统通过表名来区分不同的表
  - 修改字段的数据类型。alter table语句也可以修改字段的数据类型
  - 删除字段。删除字段是指删除已经定义好的表中的某个字段
  - 修改备注信息

【例】 在student表的Email列后面增加一列address。

```
alter table student add address varchar(30) not null after Email;
```

DDL其他操作

- 删除数据库、删除表、清空表等
  - 删除数据库语法格式如下：

```
drop database  [if exists]  db_name
```

  - 删除表、清空表的语法格式如下：

```
drop table  table_name ;
truncate table table_name ;
```

  - 操作示例

```
drop  database mysqltest; -----删除数据库
drop table `t_test_tmp` ;   ----删除表
truncate table `t_test_tmp` ;   -----清空表
```


课程目录

DML综述

- DML(Data Manipulation Language): 数据操纵语言，用于对表中的数据进行新增、删除、修改操作，分别为：
  - INSERT: 插入
  - DELETE: 删除
  - UPDATE: 更新

DML插入数据

- insert into 语句添加数据，语法格式：

```
insert [into]table_name[(col_name,...)]
 values({expr|default},...),(...),...
|set col_name ={expr|default}, ...
```

- 说明
  - 使用insert语句可以向表中插入一行数据，也可以插入多行数据
  - 一次插入多行数据，各行数据之间用“，”分隔
  - values子句包含各列需要插入的数据清单，数据的顺序要与列的顺序相对应
  - 若表名后不给出列名，则在values子句中要给出每一列的值，如果列值为空，则值必须置为null，否则会出错
  - 操作示例：

```
Insert into `dim_product`(`product_id`,`product_name`,`price`,`product_category_id`) values
('1','信封',25.83,'1'),('2','剪刀',42.25,'1');
```

DML修改数据

- update...set... 命令可以修改一个表的数据，语法格式：

```
update table_name
set col_name1= [,col_name2=expr2 ...] [where子句]
```

- 说明：
  - set子句：根据where子句中指定的条件，对符合条件的数据行进行修改。若语句中不设定where子句，则更新所有行
  - expr1、expr2.....可以是常量、变量或表达式。可以同时修改所在数据行的多个列值，中间用逗号隔开。

【例】将学号为18137221508学生的课程号为 c08106的平时成绩daily修改为80分。

```
update score set daily=80 where studentno='18137221508' and courseno='c08106';
```


DML删除数据

- 利用delete... from...语句可以从单个表中删除指定表数据，语法格式：

```
delete   [low_priority] [quick] [ignore]  from tbl_name
[where子句]  [order by子句]    [limit row_count]
```

- 说明：
  - quick修饰符：可以加快部分种类的删除操作的速度。
  - from子句：用于指定从何处删除数据。
  - where子句指定的删除条件。如果省略where子句则删除该表的所有行。
  - order by子句：各行按照子句中指定的顺序进行删除，此子句只在与limit联用时才起作用。
  - limit子句：用于告知服务器在控制命令被返回到客户端前被删除的行的最大值。

【例】删除student01表中入学成绩低于750分记录。

```
delete  from student01 where entrance <750;
```

课程目录

阿里云

DQL综述

- DQL(Data Query Language): 数据查询语言，用于查询中的数据，SELECT 语句提供了字段运算、数据筛选、分组聚合、多表关联等功能，主要涉及：SELECT、FORM、WHERE、OEDER BY、GROUP BY、HAVING 等子句。相关操作如下：

✓ 查询（SELECT）
✓ 别名 (AS)
✓ 去重（DISTINCT）
✓ 过滤（WHERE）
✓ 集合判断（IN）
✓ 范围条件 （BETWEEN AND）
✓ 模糊匹配（LIKE）
✓ 空值查询 （IS NULL）
✓ 多条件查询 （AND/OR）
✓ 排序（ORDER BY）
✓ 分组（GROUP BY & HAVING）
✓ 数量限制（LIMIT）
✓ 分支判断（CASE WHEN）

阿里云

DQL查询select

- select 语句是SQL语言从数据库中获取信息的一个基本语句：

```
select [all|distinct]selection_list
from  table_source
[where search_condition]
[group by grouping_columns][with rollup]
[having search_condition]
[order by order_expression [asc|desc]]
[limit count]
```

- 说明：
  - 可以实现从一个或多个数据库中的一个或多个表中查询信息
  - 查询结果（也称结果集）显示为另外一个二维表的形式

DQL查询select

（1）使用select语句查询一个数据表。使用select语句时，首先要确定所要查询的列。“*”代表所有的列。

【例】查询teaching数据库course表中的所有数据。

```
select * from  course;
```

（2）查询表中的指定列。针对表中的多列进行查询，只要在select后面指定要查询的列名即可，多列之间用“，”分隔。

【例】查询student表中的studentno、sname和phone数据。

```
select studentno,sname,phone from student;
```

（3）可以从一个或多个表中获取数据，即多表查询，主要采用多表连接或子查询的方式，也可以通过where子句中使用连接运算来确定表之间的联系，然后根据这个条件返回查询结果。

【例】student表、course表关联查询学生上课的信息。

```
select a.studentno,a.sname,b.cname from student a , course b where a. studentno=b. studentno;
```

DQL别名as、去重distinct

- 字段取别名，目的增加结果集的可读性，as语法格式：select项的原名 as 别名

【例】在student表中查询出生日期在2001年以后的学生的学号、姓名、电话和年龄。

```
select  studentno as '学号',sname as '姓名',phone as '手机号',year(now())-year(birthdate) as  '年龄'
from  student
where   year(birthdate)>2001;
```

- 去重 (distinct)，结果集中除去重复的行

【例】 在score表中查询期末成绩中有高于95的学生的学号和课程号,并按照学号排序。

说明：不管学生有几门课的成绩高于95，只要有一门就可以显示，利用distinct子句可将重复行消除

```
select  distinct studentno,courseno
from score
where final>95
order  by studentno;
```


DQL过滤where、集合判断in

- 使用where子句过滤结果集，过滤数据越多，返回结果集越小

【例】在student表中查询出生日期在2001年以后的学生的学号、姓名、电话和年龄。

```
select  studentno as '学号',sname as '姓名',phone as '手机号',year(now())-year(birthdate) as  '年龄'
from  student
where   year(birthdate)>2001;
```

- 带in关键字的查询,判断某个字段的值是否在于指定的集合中,相当于用or连接

【例】查询学号分别为的18135222201和18137221508的学生学号、课程号、平时成绩和期末成绩。

说明：检索条件中枚举某些确定值的范围，一般可以利用 in关键字实现

```
select studentno, courseno, daily, final
 from score
where studentno in('18135222201','18137221508');
```

DQL范围查询between and、模糊匹配like

- 带between and的范围查询，若检索条件指定排除某个范围的值，可以用 not between

【例】查询选修课程号为c05109 的学生学号和期末成绩，并且要求平时成绩在80到95之间。

```
select studentno, final
from score
where courseno='c05109'  and daily between 80 and 95;
```

- 带like的字符匹配查询
  - 使用like搜索条件时，模式字符串中的所有字符都有意义，包括开头和结尾的空格
  - like主要用于字符类型数据，字符串内的英文字母和汉字都算一个字符
  - 可用通配符并使用 not like作为查询条件
  - 通配符“%”和下划线“_”：“%”可以匹配一个或多个字符，代表任意长度字符；“_”只匹配一个字符

【例】在student表中显示所有姓何或姓韩的学生的姓名、生日和Email。

```
select  sname, birthdate, email
from student
where  sname like '何%' or sname like '韩%';
```


DQL空值查询is null、多条件查询and&or

- 用is null关键字查询空值
  - 涉及空值的查询用null来表示
  - create table语句或alter table 语句中的null表明在列中允许存在被称为null的特殊数值，不同于数据库中的其他任何值
  - 在select语句中，一个字段值是空值或者不是空值，要表示为：“is null”或“is not null”
  - 如果写成 “字段=null”或“字段<>null”，系统的运行结果都直接处理为null值，按照false处理而不报错。

【例】学生选修课程表se_course中的成绩是允许空值，以此是否成绩为空值作为查询条件，即可查到学生的选课情况。

```
select  studentno, courseno,teacherno, score
from  se_course
where score is null;
```

- 带and、or的多条件查询
  - 利用where子句指定行时，条件表达式中的字符型和日期类型值要放到单引号内，数值类型的值直接出现在表达式中

【例】在score表中显示期中高于90分、期末成绩高于85分的或者是学生学号大于100的学生学号、课程号和成绩。

```
select studentno,courseno,daily,final
from score
where  (daily >=90 and final >= 85 ) or studentno >100 ;
```

DQL排序order by

- 使用order by子句对结果集排序
  - order by子句可以对查询的结果进行（默认）升序（asc）或降序（desc）排列
  - 排序可以依照某个列的值，若列值相等则根据第二个属性的值，依此类推
  - 如果对含有null值的列进行排序时，如按升序排列，null值将出现在最前面，如按降序排列，null值将出现在最后
  - order by子句包含的列并不一定出现在选择列表中。
  - order by子句可以通过指定列名、函数值和表达式的值进行排序。
  - order by子句不可以使用text、ntext或image类型的列。
  - order by 子句中可以同时指定多个排序项。

【例】在student表中查询高于850分的学生学号、姓名和入学成绩，并按照入学成绩的降序排列。

```
select studentno 学号,sname 姓名 ,entrance 入学成绩 from student where entrance>850 order by entrance  desc;
```


DQL分组group by & having

- group by子句可以将查询结果按属性列或属性列组合在行的方向上进行分组，每组在属性列或属性列组合上具有相同的聚合值。
  - 如果聚合函数没有使用 group by子句，则只为 select语句报告一个聚合值
  - 如果将一列或多列定义成为一组，使组内所有的行在那些列中的数值相同。出现在查询的select 列表中的每一列都必须同时出现在group by子句中

【例】利用group by子句对score表数据分组，显示每个学生的学号和平均总评成绩。总评成绩计算公式如下：

总评成绩=daily*0.3+final*0.7

分析：通过学号分组，可以求出每个学生的平均总评成绩。avg()函数用于求平均值，round()函数用于对平均值的某位数据进行四舍五入。

```
select studentno 学号, round(avg(daily*0.3+final*0.7),2) as'平均分'
from score group by studentno;
```

DQL分组group by & having

- having和where都是筛选，只有那些满足搜索条件的行才用来构造结果集
  - where在select和group by 前面执行，在结果返回之前起筛选作用，不能用聚合函数或别名
  - having在select和group by 后面执行，在结果返回之后起筛选作用，可以用聚合函数或别名
  - having 子句通常与 group by 子句结合使用，也可以不带 group by, 此时having不能用聚合函数
  - 标准SQL要求having必须指定group by指定的列，或者是那些出现在聚合函数（比如sum）中的列。
  - 但是MySQL同时也支持having指定那些出现在select中的列，以及外部子查询中的列

【例】查询选课在3门以上的学生的学号及其总分，学号在100以内，查询结果按总分降序列出

分析：可以利用where筛选学号，利用having 和count 筛选分组结果，使之满足count(*)>=3的条件即可

```
select studentno 学号,sum(daily*0.3+final*0.7) as '总分'
from  score where studentno <100
group by  studentno having count(*)>=3
order by sum(daily*0.3+final*0.7) desc;
```


DQL数量限制 limit

- 用limit限制查询结果的数量
  - 可以指定查询结果从哪条记录开始显示
  - 可以指定一共显示多少条记录。limit可以指定初始位置，也可以不指定初始位置

【例】查询student表的学号、姓名、出生日期和电话，按照entrance进行降序排列，显示前3条记录。

```
select studentno,sname,birthdate,phone from  student  order by  entrance  desc  limit  3;
```

【例】查询score表中，期末成绩final高于85分的，按照平时成绩daily进行升序排列，从编号2开始，查询5条记录

```
select  *  from  score  where  final>85  order  by  daily  asc  limit  2,5;
```

说明：参数1是开始读取的第1条记录的编号（注意查询结果中，第1条记录编号为0），参数2是要查询记录的个数。

DQL分支判断

- 用case when实现分支判断，两种 case when 语法格式：

```
case case_value
   when when_value then statement_list
   [when when_value then statement_list] ...
   [else statement_list]
end case
```

```
case
   when search_condition then statement_list
   [when search_condition then statement_list] ...
   [else statement_list]
end case
```

【例】根据每个顾客的购买次数，将顾客分级。

```
select customername, ordercount,
   case ordercount
      when 1 then 'one-time customer'
      when 2 then 'repeated customer'
      when 3 then 'frequent customer'
      else 'loyal customer'
   end customertype
from cte order by customername;
```


课程目录

联结查询union

1.用UNION连接查询结果：

- UNION 操作符用于连接两个以上的 SELECT 语句的结果组合到一个结果集合中。
  - 操作语法：

    select …

    union [all | distinct] --- distinct: 可选，删除结果集中重复的数据；all则返回所有结果集，包含重复数据

    select …

    [union [all | distinct] select ...]

【例】查询sale_detail表的所有杭州地区和上海地区的记录，并将它们合并起来。

```
select * from sale_detail where region =hangzhou
union all
select * from sale_detail where region =shanghai
```

说明：操作对应的各个子查询的列个数、名称和类型必须一致。

联结查询join

2.用JOIN连接查询结果：

- ✓ inner join（内连接,或等值连接）：获取两个表中字段匹配关系的记录。
- ✓ left join（左连接）：获取左表所有记录，即使右表没有对应匹配的记录。右表不匹配的记录，右表的字段返回null。
- ✓ right join（右连接）：与 left join 相反，用于获取右表所有记录，即使左表没有对应匹配的记录。左表不匹配的记录，左表的字段返回null。
- ✓ 由于join是笛卡尔乘积计算，使用join时要注意连接字段有重复记录时，返回的记录数可能会多于输入表。

内连接 inner join　　左（外）连接 left join　　右（外）连接 right join　　交叉连接 cross join

联结查询join

【例】查询所有顾客的 ID、姓名、订单号和订单状态（顾客可能有0个或者更多订单，任意一个订单一定属于某个顾客）

```
select    c.customernumber,customername,ordernumber,status
from customers c
left join orders o using (customernumber);
```

【例】查询订单号、时间、订单行号、产品名称、销售数量、单价

```
select ordernumber,  orderdate, orderlinenumber,productname, quantityordered, priceeach
from orders
inner join orderdetails using (ordernumber)
inner join products using (productcode)
order by ordernumber, orderlinenumber;
```

【例】查询 tb_course 表中的 course_id 字段和 tb_students_info 表中的 course_id 字段相等的内容

```
select *
from tb_course
cross join tb_students_info
where tb_students_info.course_id = tb_course.id;
```


子查询

- 子查询是嵌套在另一个查询（如SELECT、INSERT、UPDATE或DELETE）中的查询：
  - 子查询可以嵌套在另一个子查询中
  - MySQL子查询称为内部查询，而包含子查询的查询称为外部查询
  - 子查询可以在使用该表达式的任何地方使用，并且必须用括号括起来

【例】查询在北京办事处工作的员工。

```
select lastname, firstname
from employees
where officecode in (
        select officecode
        from offices
        where country = 'beijing'
            );
```

关联子查询、from子查询

- 关联子查询依赖于外部查询，对外部查询中的每一行，都会计算一次关联子查询

【例】查询购买价格高于每个产品线中所有产品平均购买价格的产品。

```
select  productname,  buyprice
from products p1
where buyprice > (
            select avg(buyprice)
            from products
            where productline = p1.productline )
```

- from子句中的子查询，相当于一个临时表，此表称为派生表或物化的子查询

【例】查询销售订单中，订单物品数的最大、最小和平均数。

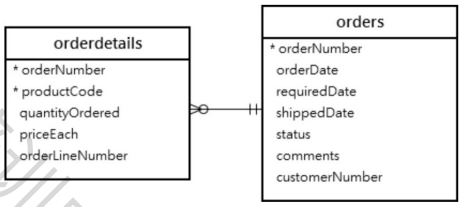
```
select   max(items), min(items), floor(avg(items))
from (   select ordernumber, count(ordernumber) as items
            from orderdetails   group by ordernumber
            ) as lineitems;
```


子查询exists&not exists

- 多种条件的组合查询
  - 当子查询与exists或not exists运算符一起使用时，子查询返回布尔值true或false
  - exists或not exists运算符通常用于关联子查询中

【例】查询至少做了一个总价值大于60K销售订单的客户。

```
select
   customernumber,
   customername
from customers
where
   exists( select
            ordernumber,
            sum(priceeach * quantityordered)
      from
         orderdetails
            inner join
         orders using (ordernumber)
      where customernumber = customers.customernumber
      group by ordernumber
      having sum(priceeach * quantityordered) > 60000);
```



课程目录

阿里云

常用函数

1 值函数
- 数学运算函数
- 字符串处理函数
- 日期处理函数

2 窗口函数
- 常见统计量类
- 排名类
- 偏移定位类
- 分组抽样类

3 聚合函数
- 常见统计量类
- 字符串类

4 其他函数
- 类型转换函数
- 分支判断函数
- 其他

阿里云

数学运算函数

三角类	整形类	运算类	随机数
• ACOS/COS -反余弦/余弦 • ASIN/SIN -反正弦/正弦 • ATAN -反正切 • COT -余切 • TAN -正切	• CEIL -向上取整 • FLOOR - 向下取整 • ROUND - 四舍五入 • TRUNCATE -截断 • CONV -进制转换	• ABS -绝对值 • LN/LOG –对数 • EXP/POW/SQRT –指数/幂/平方根 • MIN/MAX –最小/最大 • DIV/MOD –除/取模	• RAND –随机数

示例：

```
SELECT CEIL(3.1415926);
```

返回：4

字符串处理函数

长度类	查找类	整形类	正则类
• LENGTH –字符长度 • BIT_LENGTH –字符位数 • CHAR_LENGTH –字符长度	• INSTR –包含字符位置 • SUBSTR –截取字符 • SUBSTRING_INDEX –关键字截取字符串 • LEFT/RIGHT –左右截取字符串 • LOCATE –定位	• CONCAT – 连接 • LOWER/UPPER –小写/大写 • TRIM –截取空格 • LTRIM/RTRIM –左/右去空格 • LPAD/RPAD –字符串左/右填充 • FORMAT –格式化函数 • REPEAT –复制函数 • REVERSE –翻转函数	• REGEXP -正则拆分 • REGEXP_INSTR – 正则匹配指定次数的子串 • REGEXP_LIKE – 正则匹配 • REGEXP_REPLACE – 正则替换 • REGEXP_SUBSTR – 正则子串

示例：

```
SELECT concat ('ABC','123');
```

返回：ABC123

日期类型处理函数

日期获取类	日期转换类	日期运算类
• CURDATE - 当前系统日期（年月日） • CURTIME – 当前系统时间（时分秒） • DATE - 指定时间单位的日期 • DAYOFMONTH –指定时间单位的日数 • WEEKDAY - 日期值是当前周的第几天 • WEEKOFYEAR - 日期值位于当年的第几周 • NOW - 当前系统时间（年月日时分秒）	• FROM_UNIXTIME - UNIX值转换为日期 • TO_DAYS - 日期转换为天数 • TO_CHAR – 日期转换为字符串 • UNIX_TIMESTAMP - 日期转换为UNIX格式日期值 • DATE_FORMAT - 字符串转换为日期	• ADDDATE - 按指定的单位和幅度修改日期值 • DATEDIFF - 日期的差值 • SUBDATE -减去指定的时间间隔 • ADDTIME -按指定的单位和幅度修改时间值

示例：

```
SELECT dayofmonth('2021-03-01');
```

返回：1

聚合函数、其它函数

聚合函数	其它函数
• COUNT – 计数 • SUM – 求和 • AVG – 平均值 • MAX/MIN – 最大/最小值 • MEDIAN – 中位数 • STDDEV – 标准差 • STDDEV_SAMPLE – 样本标准差 • VAR_SAMP- 样本标准方差	• CAST – 类型转换 • ROW_COUNT - 返回更新的行 • CASE WHEN - 分支选择 • IFNULL – 空值判断，返回定义数值 • NULLIF – 数值判断，返回空值 • USER - 客户端提供的用户名和主机名 • CONNECTION_ID - 返回连接ID（线程ID） • DATABASE - 返回默认（当前）数据库名称

示例：

```
SELECT COUNT(*) FROM tmp;
```

返回：表tmp行数

窗口函数（MySQL 8.0）

统计类	排名类	其它类
• COUNT – 计数 • SUM – 求和 • AVG – 平均值 • MAX/MIN – 最大/最小值 • MEDIAN – 中位数 • STDDEV – 标准差 • STDDEV_SAMPLE – 样本标准差	• ROW_NUMBER – 行号 • RANK – 跳跃排名 • DENSE_RANK – 连续排名 • PERCENT_RANK – 相对排名 • CUME_DIST – 值中值的累积分布	• LAG - 按偏移量取之前第几行 • LEAD - 按偏移量取之后第几行 • FIRST_VALUE– 首行记录 • LAST_VALUE – 尾行记录 • NTH_VALUE – 窗口中第几行 • NTILE – 记录分成指定组，组序号一致

示例：

```
SELECT score
,ROW_NUMBER() OVER (PARTITION BY class_id ORDER BY
score DESC) AS rk
FROM table_score;
```

table_score

class_id	score
1	99
1	98
2	97
2	96

返回：

score	rk
99	1
98	2
97	1
96	2

阿里云

阿里云全球培训中心

阿里云全球培训中心

数据分析工具

阿里云大数据分析师ACP认证课程

阿里云

课程目标

学习完本课程后，你将能够：

1. 了解常见数据分析工具
2. 了解阿里云大数据分析平台基础知识
3. 掌握阿里云大数据分析平台应用

阿里云

阿里云全球培训中心

课程目录

1. **常见分析工具简介**
2. 阿里云大数据平台简介
3. 阿里云大数据平台应用

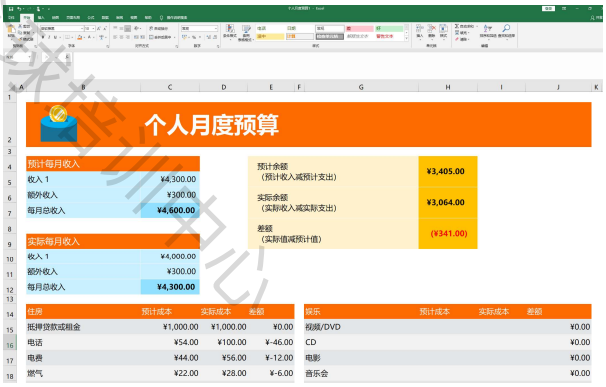
数据分析常见工具

- “工欲善其事，必先利其器”，适宜的分析工具极大提升数据分析工作效率，常见分析工具有：
  - EXCEL
  - Python
  - R
  - Spss
  - 阿里云机器学习平台PAI
  - 阿里云大数据分析平台MaxCompute+DataWorks

数据分析常见工具-EXCEL

- Excel作为入门级的工具，是最基础也是最主要的数据分析工具

  - 提供多种数据分析工具，如模拟运算表、单变量求解、规划求解、方案管理器等
  - 技术门槛低，上手快，无需编程；
  - 常用操作有函数、可视化、数据透视表、VBA等；
  - 不能对用户进行角色管理，保密性较差；
  - 不能处理大数据，数据量过大，查询和计算效率低；
  - 图形展现不太灵活易用，包括颜色、字体的设定等；



数据分析常见工具-Python

- Python是一种面向对象、解释型计算机程序设计语言；常用于数据分析和交互、探索性计算以及数据可视化等。

简单易学 | 成熟活跃

高效开发 | 胶水语言

常用Python 库:

- ✓ **NumPy**是Python进行数据分析时所用到的一个基础库，提供了矩阵运算的功能和各种数学函数，是其他的数据处理库的基础；
- ✓ **SciPy**是一个开源的Python算法库和数学工具包，包含的模块有最优化、线性代数、积分、插值、特殊函数、快速傅里叶变换、信号处理和图像处理、常微分方程求解和其他科学与工程中常用的计算；
- ✓ **Matplotlib**是一个Python 2D绘图库，只需几行代码即可生成绘图，直方图，功率谱，条形图，错误图，散点图等；
- ✓ **Pandas**基于NumPy的一个专门用于数据分析的开源的Python库。用于分组统计、合并数据、数据筛选、以及时间序列操作；
- ✓ **Scikits-learn** 是用于处理复杂数据的最优秀的机器学习库之一；包含大量传统机器学习、数据挖掘任务的算法，比如数据降维、分类、回归、聚类、以及模型选择等。

数据分析常见工具-R

- R是一个全面的统计研究平台，提供了各式各样的数据分析技术
  - 几乎任何类型的数据分析工作皆可在R中完成
  - R是用于统计学计算和绘图的语言
  - R作为开源的项目，有很多人做了扩展包，使得统计绘图和分析更加简单

常见算法库示例：

✓ 聚类：

基于划分的方法: kmeans, pam, pamk, clara 基于层次的方法: hclust, pvclust, agnes, diana 基于模型的方法: mclust 基于密度的方法: dbscan 等；

✓ 分类：

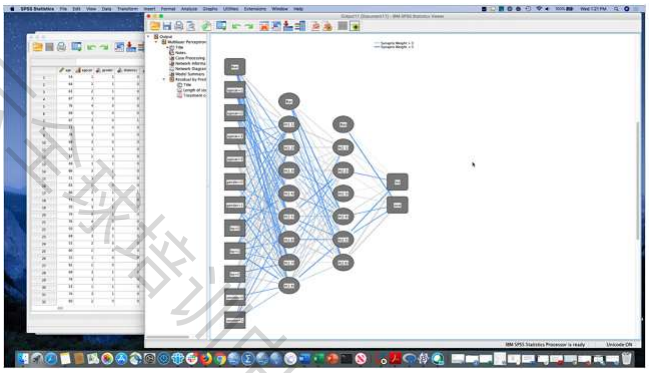
决策树: rpart, ctree 随机森林: cforest, randomForest 回归, Logistic回归, Poisson回归: glm, predict, residuals等；

✓ 统计分析

方差分析: aov, anova 密度分析: density 假设检验: t.test, prop.test, anova, aov 线性混合模型：lme 主成分分析和因子分析：princomp等

数据分析常见工具-SPSS

- SPSS采用类似Excel表格的方式输入与管理数据，数据接口较为通用，能方便地从其他数据库中读入数据。突出特点是操作界面友好，输出结果美观：
  - 操作简单，易于使用
  - 高效数据整合
  - 快速可靠，自动化数据准备
  - 功能强大、算法丰富、灵活的功能模块组合
  - 开源集成、通过扩展增强 SPSS 语法

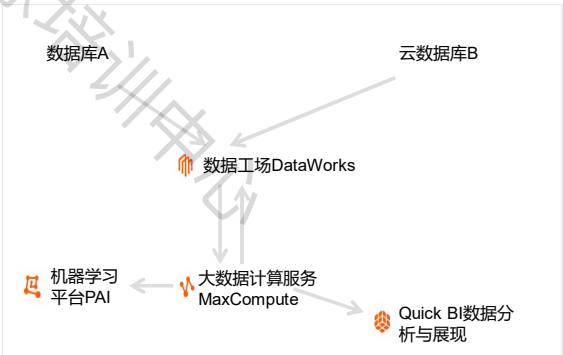


数据分析常见工具-阿里云大数据分析平台

阿里云大数据计算服务MaxCompute配合**DataWorks**（数据工场）构建的大数据分析平台，首先将数据同步至MaxCompute，通过SQL或其他技术进行分析与处理。

平台优势：

- ✓ 提高工作效率：将数据同步至MaxCompute，通过SQL语句可以进行分析与处理，提高工作效率；
- ✓ 降低大数据分析使用门槛：MaxCompute提供多种开源软件的插件，可以轻松完成数据上云、进行数据分析；
- ✓ 快速响应业务需求：可以根据业务的数据分析需求，快速灵活地进行响应与满足；
- ✓ 提升业务洞察能力：通过MaxCompute计算能力，可以实现针对海量用户的精细化运营。



数据分析常见工具-阿里云机器学习平台PAI

机器学习PAI

阿里云机器学习平台PAI（Platform of Artificial Intelligence**）：**是构建在阿里云MaxCompute计算平台之上，集数据处理、建模、离线预测、在线预测为一体的机器学习平台。为算法开发者提供了丰富的MPI、PS、BSP等编程框架和数据存储接口，同时提供了基于WEB的可视化控制台，降低了使用门槛。

可视化操作界面

通过对底层分布式算法的封装，提供拖、拉、拽的可视化操作环境。使机器学习的创建过程像堆积木一样简单。

算法丰富

提供特征过程、数据预处理、统计分析、机器学习、深度学习框架、预测评估等几百种算法组件。

一站式体验

提供模型训练功能，还提供在线预测功能以及离线调度功能，让机器学习训练结果和业务可以无缝衔接。

多种计算框架

流计算Flink、深度学习TensorFlow、大规模并行计算Parameter Server以及Spark、PySpark、MapReduce等主流框架。

课程目录

阿里云

大数据计算服务MaxCompute

MaxCompute

大数据计算服务（MaxCompute）是一种快速、完全托管的EB级数据仓库解决方案。

- 致力于批量结构化数据的存储和计算，提供海量数据仓库的解决方案及分析建模服务。
- 适用于100GB以上规模的存储及计算需求，最大可达EB级别。
- 支持SQL、MapReduce、UDF（Java/Python）、Graph、基于DAG的处理、交互式、内存计算、机器学习等计算类型及MPI迭代类算法。简化了企业大数据平台的应用架构。

分布式

分布式集群架构
跨集群技术突破
集群规模可扩展

安全性

自动存储容错
安全沙箱机制
冗余存储机制

易用性

标准接口服务
数据上传下载
支持SQL语句

管理与授权

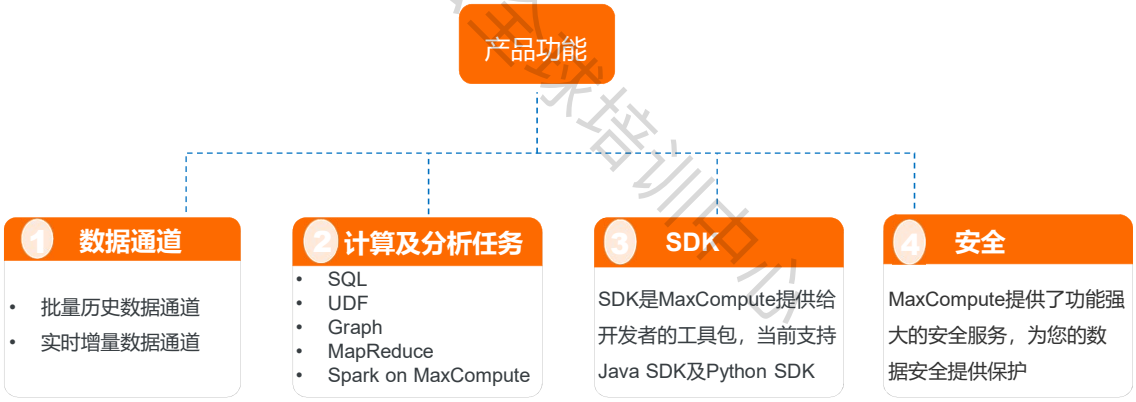
多用户的管理协同
多种方式权限管理
灵活访问控制策略

低成本

免运维
弹性伸缩
灵活计费模式

阿里云

MaxCompute的功能概述



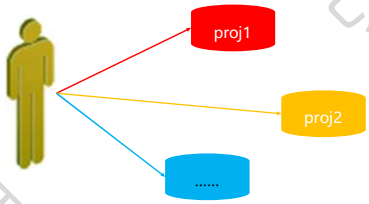
MaxCompute的数据组织

MaxCompute基本概念

项目空间
Project

项目空间（Project）：是 MaxCompute 的基本组织单元，它类似于传统数据库的 Database 或 Schema 的概念，是进行**多用户隔离**和**访问控制**的主要边界。

- ✓ 一个用户可以同时拥有多个项目空间的权限；
- ✓ 通过安全授权，可以在一个项目空间中访问另一个项目空间中的对象，例如：表、资源、函数、实例等；
- ✓ 使用命令“use project ” 进入指定空间。



MaxCompute基本概念

表
Table

表（Table）：表是 MaxCompute 的数据存储单元，它在逻辑上也是由行和列组成的二维结构，每行代表一条记录，每列表示相同数据类型的一个字段，一条记录可以包含一个或多个列，各个列的名称和类型构成这张表的 Schema。

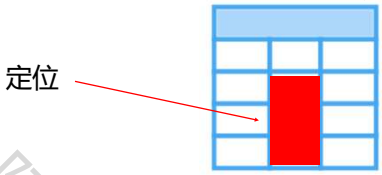
- ✓ MaxCompute 中不同类型计算任务的**操作对象（输入、输出）都是表。**
- ✓ 可以创建表、删除表以及向表中导入数据。
- ✓ 表分为**内部表**和**外部表。**

MaxCompute基本概念

分区 Partition

分区（Partition）： 是指在创建表时指定分区空间，即指定表内的某几个字段作为分区列。

- ✓ 可以将分区类比为文件系统下的目录,MaxCompute 将分区列的每个值作为一个分区（目录）；
- ✓ 可以指定多级分区，即将表的多个字段作为表的分区，分区之间正如多级目录的关系；
- ✓ 指定分区使用数据时，可避免全表扫描，提高处理效率，降低费用。



MaxCompute基本概念

生命周期 Lifecycle

生命周期（Lifecycle）： 指表（分区）数据从最后一次更新的时间算起，在经过指定的时间后没有变动，则此表（分区）将被 MaxCompute 自动回收。这个 **指定的时间** 就是生命周期。

- ✓ 生命授权单位：days（天），只接受正整数且设定级别为表；
- ✓ 分区不设定默认为永久，则表（分区）不会根据生命周期规则被MaxCompute自动回收；
- ✓ 生命周期只能设定到表级别，不能在分区级设置生命周期；
- ✓ 创建表时即可指定生命周期。

MaxCompute基本概念

资源
Resource

资源（Resource）： 是 MaxCompute 的特有概念，使用UDF或MR功能需要通过资源来实现。

资源类型：

- File类型
- Table类型
- Jar类型
- Archive 类型

MaxCompute基本概念

任务
Task

任务（Task）： 是 MaxCompute 的基本计算单元。

- ✓ 单个 SQL Query 或者 MapReduce 程序统称为一个任务。
- ✓ 执行计划逻辑上可以被看做一个**有向图**，图中的点是执行阶段，各个执行阶段的依赖关系是图的边。
- ✓ 同一个执行阶段内，会有多个进程Worker，共同完成计算工作。
- ✓ 同一个执行阶段的不同 Worker 只是处理的数据不同，执行逻辑完全相同。

注意：在MaxCompute中，并不是所有的请求都会被转化为任务（Task），如对项目空间、资源、实例、自定义函数的操作。

MaxCompute基本概念

实例 Instance

实例（Instance）： 在 MaxCompute 中部分任务执行时会实例化。

- ✓ 实例会经历**运行**（Running）和**结束**（Terminated）两个阶段。
- ✓ **运行**中的实例状态为 Running（运行中）；
- ✓ **结束**阶段则会有 Success（成功），Failed（失败）或Canceled（被取消）三种状态；
- ✓ 任务每调度执行一次就形成一个实例
- ✓ 停止指定的实例，实例状态必须为Running

MaxCompute基本概念

函数 Function

函数（Function）： 在 MaxCompute 中封装的计算功能。可以分为两大类：内部函数和自定义函数。

自定义函数又分为三种：

- ✓ UDF自定义标量函数
- ✓ UDAF自定义聚合函数
- ✓ UDTF自定义表值函数

课程目录

DataWorks是什么

DataWorks（数据工场）是阿里云重要的PaaS平台产品，它提供全面托管的工作流服务，一站式开发管理的界面，帮助企业专注于数据价值的挖掘和探索。

全面托管的调度	多种任务类型	可视化开发	监控告警
具有强大的调度能力 提供完全托管的服务	数据同步、 ODPS SQL、 MR、 SHELL、机器学习	提供可视化的代码开发、 工作流设计器界面	可视化的任务监控 任务监控短信报警

DataWorks核心功能

数据集成：主要用于离线/实时数据同步。提供复杂网络环境下、丰富的异构数据源之间高速稳定的数据移动及同步能力；

数据开发：对MaxCompute中的数据进行加工（MaxCompute SQL、MaxCompute MR）、分析与挖掘（数据分析、数据挖掘）等处理，从而发现其价值；

数据应用： MaxCompute中的数据经加工处理后，应用各种场景，如数据提取、数据交互、数据报表、数据分析；

数据服务：提供为企业搭建统一的数据服务总线，帮助企业统一管理对内对外的API服务。提供快速将数据表生成API的能力，同时支持快速注册现有的API至数据服务平台，进行统一的管理和发布；

数据治理：多种数据处理视角下的数据治理流程与工具引导，满足数据生产、数据使用、数据管理场景下的主要数据治理需求，如数据质量、数据地图、安全中心、数据保护伞；

任务运维：各种数据处理任务的发布、监控、运维。

DataWorks基本概念

1 计算引擎（MaxCompute Project）

2 工作空间（DataWorks Workspace）

3 任务/实例（Task / Instance）

4 业务流程

5 解决方案

6 组件

7 输出名称

8 脚本开发 /提交

术语

计算引擎

DataWorks支持多种计算和存储引擎服务，包括离线计算MaxCompute（最常见）、开源大数据引擎E-MapReduce、实时计算（基于Flink）、机器学习PAI、图计算服务Graph Compute和交互式分析服务等，并且支持用户自定义接入计算和存储服务。

工作空间

工作空间是大数据开发平台DataWorks最基本的组织对象，是进行多组织隔离和访问控制的主要边界；也是用户管理表（Table）、资源（Resource）、自定义函数（UDF）、节点（Node）、工作流（WorkFlow）、权限等的基本单元。

在大数据开发平台中，一个DataWorks项目空间对应绑定一个或多个MaxCompute project。

术语

任务

任务是对数据执行的操作。每个任务使用0或0个以上的数据表（数据集）作为输入，生成一个或多个数据表（数据集）作为输出。任务主要分为节点任务（Node Task）、工作流任务（Flow Task）和内部节点（inner Node）。

实 例

在DataWorks中，节点任务在执行时会被实例化，并以MaxCompute实例的形式存在。实例会经历未运行、等待时间/等待资源、运行中、成功/失败几个状态。

当天晚上23：30节点转实例！

术语

业务流程

针对业务实体，抽象出了业务流程的概念，能够以业务视角来组织数据代码开发，提升任务管理效率。一个业务流程可以被多个解决方案复用。支持多级子目录（4级）、可按业务流程组织、发布，提供基于任务类型的组织方式。

解决方案

解决方案是自定义组合一些业务流程。可以包含多个业务流程，解决方案之间可复用相同的业务流程。

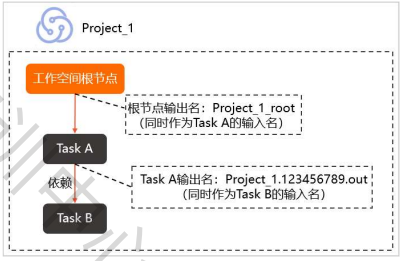
术语

组件

是一种带有多个输入参数和输出参数的SQL代码过程模板，SQL代码的处理过程一般是引入一到多个源数据表，通过过滤、连接、聚合等操作，加工出新的业务需要的目标表。将SQL中的通用逻辑抽象成为组件，提升代码复用性。

输出名称

每个任务（Task）输出点的名称，它是在单个租户（阿里云帐号）内设置依赖关系时，用于连接上下游两个任务（Task）的虚拟实体。当在设置某任务与其他任务形成上下游依赖关系时，必须根据**输出名称**（而不是节点名称或节点ID）来完成设置，设置完成后该任务的输出名也同时作为其下游节点的输入名称。



术语

脚本开发

脚本开发是提供给数据分析使用的一个代码存储空间。脚本开发的代码无法发布到调度系统，无法进行调度参数配置，仅可以进行部分数据查询分析的工作。

提交

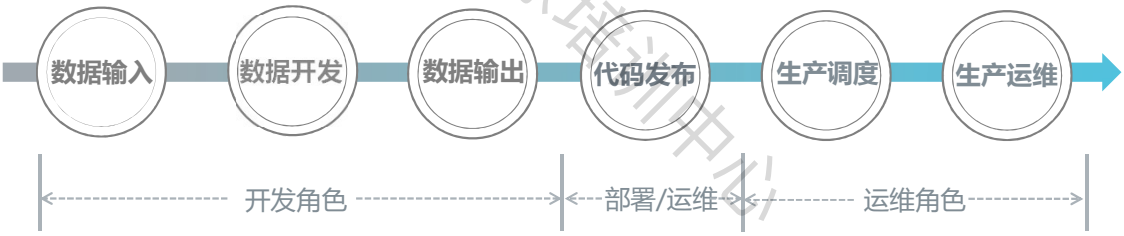
提交是指开发的节点任务、业务流程，从DataWorks开发环境发布至调度系统的过程。完成提交后，相应的代码、调度配置全部合并至调度系统中，调度系统根据相关配置进行调度操作。

课程目录

应用流程

涉及模块：

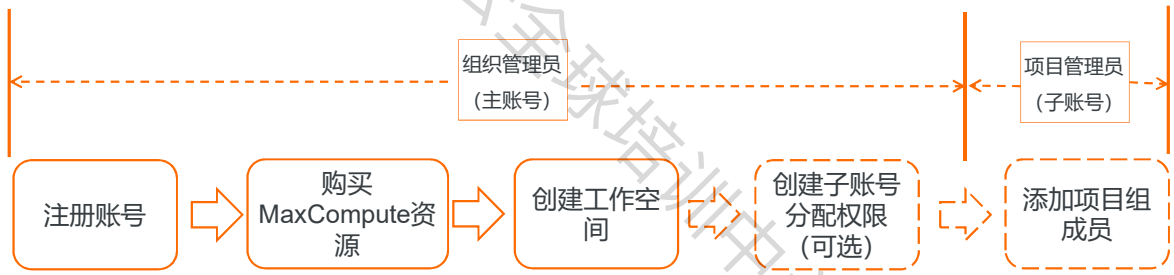
数据集成模块　数据开发模块　发布管理模块　运维中心模块



基于数据平台进行数据分析的简单流程。

课程目录

服务开通



服务开通

阿里云官网https://cn.aliyun.com/产品栏搜索MaxCompute

立即开通

商品类型

- 新用户，初期使用，消耗资源少，适合按量付费

区域

- 考虑与阿里云上其他产品关系

规格类型

- 评估所需计算资源和存储资源选择合适规格

计算资源单元

- 如果规格选择标准计算资源，要选择预留计算资源CU
- 如果规格选其他套餐，则为套餐包含的CU

非预留计算资源

- 不独享，但保证全天全天50%的计算力

创建项目空间

- 通过MaxCompute控制台或DataWorks控制台创建MaxCompute项目，两种操作功能一致；但是由于MaxCompute控制台的项目管理和查询编辑功能由DataWorks实现，因此创建MaxCompute项目时，会先创建DataWorks工作空间。

前提条件:

- 阿里云账号或RAM用户已开通DataWorks服务和MaxCompute服务且服务开通区域一致；
- 如使用RAM用户账号登录阿里云官网和创建项目，首先确认账号可用并已授权。

创建项目空间

- 通过MaxCompute控制台或DataWorks控制台创建MaxCompute项目，两种操作功能一致；但是由于MaxCompute控制台的项目管理和查询编辑功能由DataWorks实现，因此创建MaxCompute项目时，会先创建DataWorks工作空间。

前提条件:

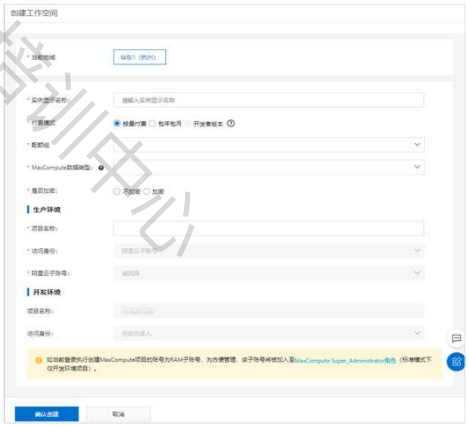
- 阿里云账号或RAM用户已开通DataWorks服务和MaxCompute服务且服务开通区域一致；
- 如使用RAM用户账号登录阿里云官网和创建项目，首先确认账号可用并已授权。

创建项目空间

- 通过MaxCompute控制台或DataWorks控制台创建MaxCompute项目，两种操作功能一致；但是由于MaxCompute控制台的项目管理和查询编辑功能由DataWorks实现，因此创建MaxCompute项目时，会先创建DataWorks工作空间。

前提条件:

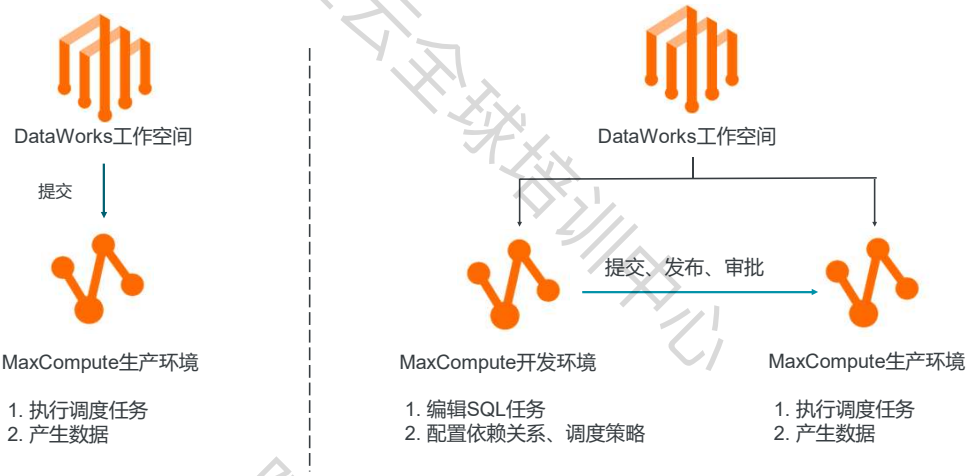
- 阿里云账号或RAM用户已开通DataWorks服务和MaxCompute服务且服务开通区域一致；
- 如使用RAM用户账号登录阿里云官网和创建项目，首先确认账号可用并已授权。



创建工作空间

工作空间是DataWorks管理任务、成员，分配角色和权限的基本单元，所以首先创建工作空间：

简单模式 vs 标准模式



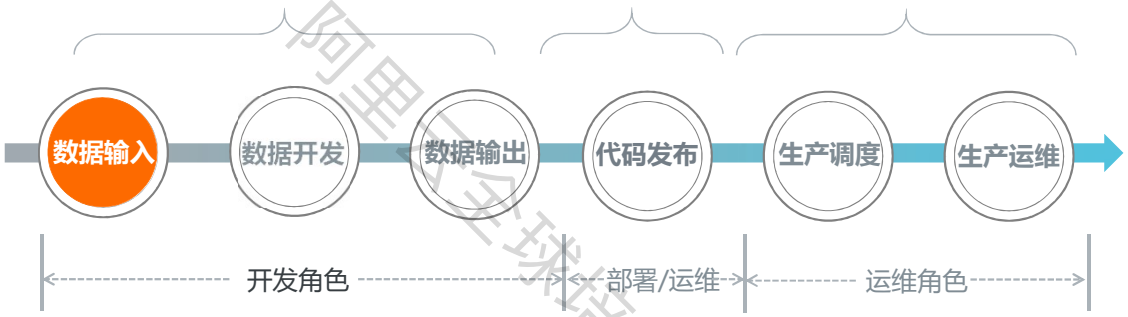
添加项目空间成员

组织管理员创建子账号并添加为组织成员

课程目录

应用流程

涉及模块：**数据集成模块** 数据开发模块 发布管理模块 运维中心模块

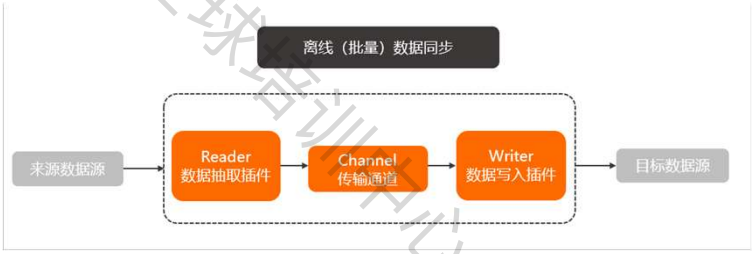


数据输入即数据同步：将分析所需数据导入到大数据平台。

数据输入-数据集成

数据集成是稳定高效、弹性伸缩的数据同步平台，致力于提供复杂网络环境下、丰富的异构数据源之间高速稳定的数据移动及同步能力：

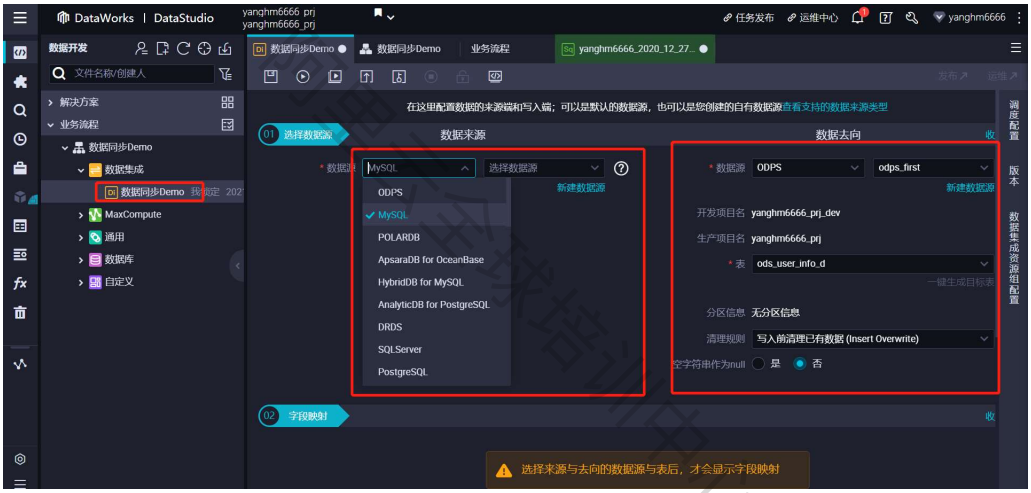
- ✓仅支持结构化数据同步；
- ✓支持单地域内及部分跨地域的相互同步、交换；
- ✓仅完成数据同步（传输），本身不提供数据流的消费方式



阿里云

数据输入-数据集成配置界面

DataWorks数据集成可实现各种数据源与MaxCompute之间数据同步



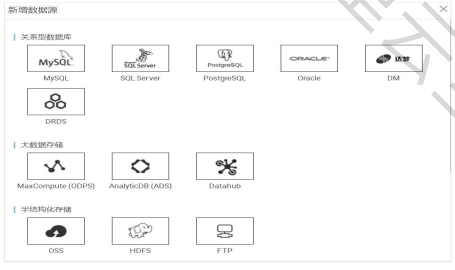
阿里云

数据输入-同步数据流程

数据输入-配置数据源

配置数据源步骤：

1. 选择新增数据源类型；
2. 配置数据源相关属性：如数据源名称、连接信息、登录名以及密码（不同的数据源配置的属性信息不同）；
3. 测试连通性，目的是验证输入的数据源信息是否正确；
4. 确定，完成配置。



选择数据源类型

配置数据源类型

注：在数据开发过程中，需由**项目管理员**在【项目管理>数据源配置】来新增数据源供开发使用。

数据输入-创建数据同步任务

在 DataWorks中，通常使用数据集成功能，将业务系统中产生的业务数据定期导入到工作区，通过工作流任务的加工处理后，再将计算结果定期导出到指定的数据源中，供进一步展示或使用。

同步任务创建步骤：

1、新建同步任务（节点）
2、选择来源
3、选择目标
4、字段映射
5、通道控制
6、调度配置
7、保存提交

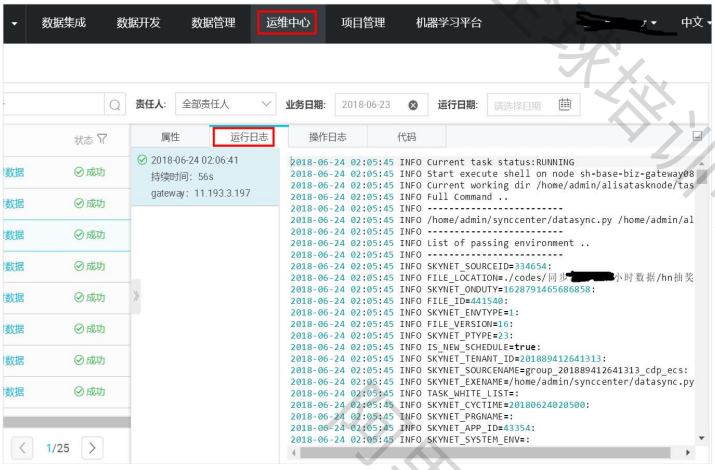
数据输入-运行数据同步任务

数据同步任务的运行方式有：

直接运行	调度运行	补数据运行	测试节点运行
• 指运行在默认的资源组上 • 可以在当前窗口下方直接看到运行日志 • 日志不保存，消失后不可见	• 基于调度系统配置调度任务 • 一般在第二天产生运行实例 • 节点到设置调度时间开始运行任务。	• 一般是运行失败同步任务或批量处理历史任务数据 • 手动操作执行	• 测试节点选择当天以前的实例节点运行 • 任务立刻运行并查看日志。

数据输入-查看数据同步任务结果

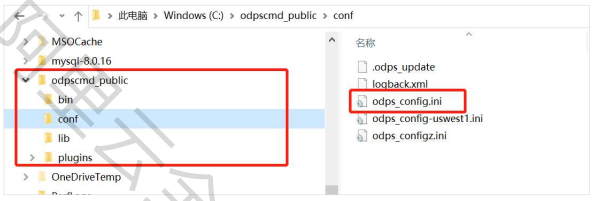
查看同步任务结果：通过DataWorks的运维中心查看同步任务日志，日志中包含了同步处理的详细信息，如果发生错误，可以根据日志进行排查处理。



数据输入-客户端方式

通过MaxCompute客户端同步数据，客户端是一个java程序，需要JRE环境才能运行，请下载并安装JRE 1.8+版本，安装步骤如下：

1. 下载客户端，解压



2. 修改配置文件odps_config.ini

```
project_name=yanghm6666_prj
access_id=LTAI4                    iLXAyCVXBcc
access_key=gkTw                    Hz9aAPt4Ip84
end_point=http://service.cn-hangzhou.maxcompute.aliyun.com/api
tunnel_endpoint=http://dt.odps.aliyun.com
log_view_host=http://logview.odps.aliyun.com
```

3. 启动运行客户端

```
C:\Users\13911>odpscmd
Aliyun ODPS Command Line Tool
Version 0.30.2
@Copyright 2018 Alibaba Cloud Computing Co., Ltd. All rights reserved.
odps@ yanghm6666_prj>
```

提示：为了运行方便，可以把客户端bin目录添加到系统PATH变量。

数据输入-客户端Tunel命令上传、下载数据

- 通过 tunnel上传数据：
  - 支持文件或目录(指一级目录)的上传
  - 每一次上传只支持数据上传到一个表或表的一个分区
  - 有分区的表一定要指定上传的分区，多级分区一定要指定到末级分区

执行命令示例： tunnel upload d:\test\p1.txt test_01.t_test_p/gender='male';

- 通过 tunnel下载数据：
  - 只支持下载到单个文件
  - 每一次下载只支持一个表或者一个分区
  - 有分区的表一定要指定下载的分区，多级分区一定要指定到末级分区

执行命令示例： tunnel download test_01.t_test_p/gender='male' d:\test\p_download.txt

数据输入-DataWorks上传本地文件

DataWorks支持上传本地的文本文件至工作空间的表中

本地文本件上传的限制：

- ✓ 文件类型：仅支持.txt、.csv和.log类型的文件
- ✓ 文件大小：不能超过30 MB
- ✓ 操作对象：支持分区表导入和非分区表导入，但不支持分区值为中文

课程目录

应用流程

涉及模块：数据集成模块　**数据开发模块**　**发布管理模块**　运维中心模块

数据开发：数据同步到MaxCompute后，在DataWorks开发业务流程，实现数据加工处理逻辑。测试后，将开发的流程代码打包，交由部署/运维人员发布。

DataWorks数据开发总览

数据开发模式采用**项目**>**解决方案**>**业务流程**三级结构，按照业务种类将相关的不同类型的节点任务组织在一起。

DataWorks开发体系：

- **项目**：权限组织的基本单位，用来控制用户的开发、运维等权限。
- **解决方案**：可自定义组合一些业务流程为一个解决方案。一个方案可以包含多个流程；解决方案之间可以复用相同的流程；组织完成的解决方案可以沉浸式开发。
- **业务流程**：对业务的抽象实体。以业务的视角来组织数据代码开发，业务流程可以被多个解决方案复用。

建表

- DataWorks提供了DDL模式、向导模式构建MaxCompute表

数据开发流程

DataWorks中，ODPS SQL节点、Shell节点、PyODPS节点等各类节点的开发过程大同小异，根本区别在于个不同类型节点的数据处理实现。

ODPS SQL节点开发过程示例如下：

数据开发工具- MaxCompute SQL

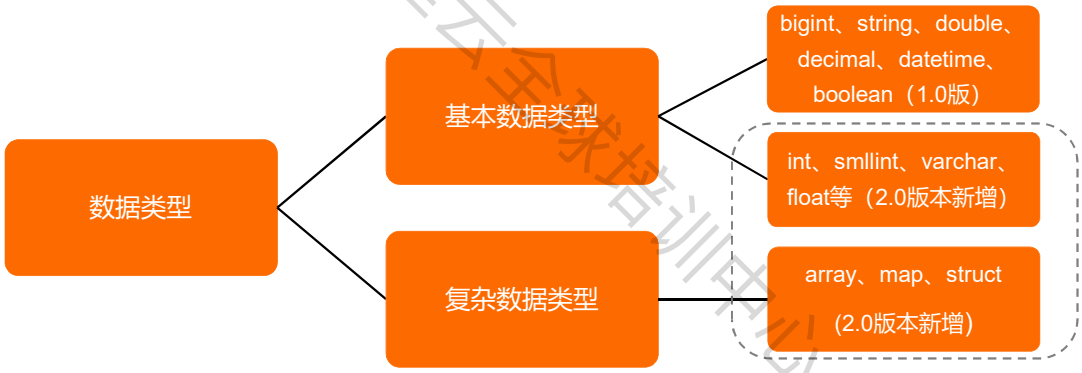
- MaxCompute SQL
  - 是个数据仓库工具
  - 采用的是类似于SQL的语法，是标准语法ANSI SQL92的一个子集，并有自己的扩展
  - 支持常用的 SQL 语法，包括窗口函数
  - 适用于海量数据(TB/PB级别)的数据运算

示例

```
INSERT INTO TABLE t_dml_p
SELECT province
,city
,product_id
,sum(cnt)
,sum(amt)
FROM t_dml
GROUP BY province
,city
,product_id
;
```

- MaxCompute SQL特征
  - 不等同于关系数据库，不支持事务、主外键约束、索引，也不支持Update、Delete等语法
  - Query 解析后，计算和存储交由底层的飞天实现
  - 任务准备、提交等阶段要花费较长时间，实时性不高
  - SQL 长度有限制，目前是不能超过 2MB

数据开发工具-MaxCompute SQL的数据类型



使用2.0版新增数据类型时，需要进行参数设置：

```
set odps.sql.type.system.odps2=true;
set odps.sql.decimal.odps2=true;
```

使用Hive兼容版本数据类型时，需要进行参数设置：

```
set odps.sql.type.system.odps2=true;
set odps.sql.decimal.odps2=true;
set odps.sql.hive.compatible=true;
```


数据开发工具-MaxCompute SQL快速建表CTAS

建表基本语法：

```
CREATE TABLE [IF NOT EXISTS] table_name
[(col_name data_type [COMMENT col_comment],…)]
[COMMENT table_comment]
[PARTITIONED BY (col_name data_type [COMMENT col_comment],…)]
[LIFECYCLE days]
[AS select_statement];
```

使用LIKE 和 AS快速建表：

```
CREATE TABLE <table_name> AS SELECT <column_list> FROM <table_name> WHERE …;
CREATE TABLE <table_name> LIKE <table_name>;
```

二者区别：

- **数据：**
  - AS 可以带入数据，可以依赖于多张表
  - LIKE 只能复制单张表的表结构，不能带入数据
- **属性：**
  - AS 不能带入LIFECYCLE、分区键信息、注释等
  - LIKE 不能带入LIFECYCLE，可以带入分区键信息、注释等

数据开发工具-MaxCompute SQL分区操作 Partition

添加分区：

```
ALTER TABLE <table_name> ADD [IF NOT EXISTS] PARTITION <pt_spec> [PARTITION <pt_spec>
PARTITION<pt_spec>...];
pt_spec: (partition_col1=partition_value1, partition_col2=partition_value2,…)
```

删除分区：

```
ALTER TABLE <table_name> DROP [IF EXISTS] PARTITION <partition_filtercondition>;
```

```
partition_filtercondition
   : partition (<partition_col> <relational_operators> <partition_col_value>)
   | partition (scalar(<partition_col>) <relational_operators> <partition_col_value>)
   | partition (<partition_filtercondition1> AND|OR <partition_filtercondition2>)
   | partition (NOT <partition_filtercondition>)
   | partition (<partition_filtercondition1>)[,partition (<partition_filtercondition2>), ...]
```


数据开发工具-MaxCompute SQL表的生命周期操作

创建表时指定表的生命周期

```
CREATE TABLE <table_name> (<col_spec>) LIFECYCLE days ;
```

修改表的生命周期

```
ALTER TABLE <table_name> SET LIFECYCLE days ;
```

禁止或恢复生命周期

```
ALTER TABLE <table_name> DISABLE LIFECYCLE ;
ALTER TABLE <table_name> ENABLE LIFECYCLE ;
```

修改更新时间

```
ALTER TABLE <table_name> TOUCH PARTITION (<pt_spec>);
```

数据开发工具-MaxCompute SQL的数据更新

基本语法：输出到普通表或者静态分区

```
INSERT {INTO|OVERWRITE} TABLE <table_name> [PARTITION (<pt_spec>)] [(<col_name> [,<col_name> ...)]
<select_statement>
FROM <from_statement>
```

基本语法：输出到动态分区

```
INSERT {INTO|OVERWRITE} TABLE <table_name> PARTITION (<ptcol_name>[, <ptcol_name> ...])
<select_statement> FROM <from_statement>;
```

- 如果目标表有多级分区，在运行 insert 语句时允许指定部分分区为静态，但是静态分区必须是高级分区
- 动态生成的分区值不可以为 NULL

数据开发工具-MaxCompute SQL的MULTI INSERT

基本语法：

```
FROM <from_statement>
INSERT OVERWRITE | INTO TABLE <table_name1> [PARTITION (<pt_spec1>)]
<select_statement1>
INSERT OVERWRITE | INTO TABLE <table_name2> [PARTITION (<pt_spec2>)]
<select_statement2>
...;
```

多路输出的限制：

- 单个SQL里最多可以写 255 路输出
- 对于分区表，同一个目标分区不可以出现多次
- 对于未分区表，该表不能作为目标表出现多次
- 对于同一张分区表的不同分区，不能同时有 insert overwrite 和 insert into 操作

数据开发工具-MaxCompute SQL 的MAPJOIN

使用情景：

一个大表和一个或多个小表做 join

基本原理：

将用户指定的小表全部加载到执行 join 操作的程序的内存中，从而加快 join 的执行速度

注意事项：

- left outer join 的左表必须是大表
- right outer join 的右表必须是大表
- inner join 左表或右表均可以作为大表
- full outer join不能使用 mapjoin
- 支持小表为子查询，需要使用别名
- 可以使用不等值连接或者使用 or 连接多个条件
- 目前最多支持指定128张小表
- 所有小表加载到内存后占用的内存总和不得超过512M

示例1：不等值 or 作为连接条件

```
SELECT /*+ MAPJOIN(a) */
a.shop_name
,a.total_price
,b.total_price
FROM sale_detail_sj a
JOIN sale_detail b
ON a.total_price < b.total_price
OR a.total_price + b.total_price < 500;
```

示例2：提升性能

```
SELECT /* + MAPJOIN(a) */
a.shop_name
,b.customer_id
,b.total_price
FROM shop a
JOIN sale_detail_b
ON a.shop_name = b.shop_name;
```


数据开发工具-其它

数据开发工具-用户自定义函数

用户自定义函数（User Defined Function，UDF），除了使用内置函数外，用户可以通过创建自定义函数来满足不同的计算需求。

UDF分类	描述
UDF（User Defined Scalar Function）	用户自定义标量值函数。其输入与输出是一对一的关系，即读入一行数据，写一条输出值 。
UDTF（User Defined Table Valued Function）	自定义表值函数。用来解决一次函数调用输出多行数据的需求。它是唯一能够返回多个字段的自定义函数。UDTF不等于UDT（User Defined Type）
UDAF（User Defined Aggregation Function）	自定义聚合函数。其输入与输出是多对一的关系， 即将多条输入记录聚合成一条输出值。它可以与SQL中的GROUP BY语句联用。

数据开发工具-UDF 开发使用流程

UDF流程

开发打包

使用 Java SDK 开发测试UDF，导出Jar包

上传 Jar 包

将导出的 jar 包上传到 MaxCompute

创建函数

在 MaxCompute 中基于 jar 包新建函数

测试使用

测试新建的函数，并在合适的情况下调用

数据开发工具-MapReduce

- **MapReduce** 是由 Google 最早提出的分布式数据处理模型，MR是一种简单易用编程框架
  - 基于MR框架的程序可以运行在多个机器的集群上，并以一种安全可靠的方式处理TB、PB 级别的海量数据
  - **MR** 框架处理数据过程主要分成两个阶段：Map 阶段和 Reduce 阶段
  - **MR** 框架运转在<key, value> 键值对上，框架把作业的输入看为是一组<key, value> 键值对，同样也产出一组 <key, value> 键值对做为作业的输出，这两组键值对的类型可能不同。
  - MR处理适合半结构和非结构化海量数据的批处理
  - MR处理性能可随机器数量线性扩展
  - MR框架适用多种应用场景，如Web访问日志分析
  - 可应用于机器学习的复杂算法

MapReduce 编程模型

- MaxCompute提供三个版本的MapReduce编程接口：
  - MaxCompute MapReduce：MaxCompute的原生接口，执行速度快、开发快捷、不暴露文件系统。
  - MaxCompute扩展MapReduce（MR2）：对MaxCompute MapReduce的扩展，支持更复杂的作业调度逻辑，MapReduce的实现方式与MaxCompute原生接口一致。
  - Hadoop兼容版本MapReduce：高度兼容Hadoop MapReduce，与MaxCompute MapReduce MR2不兼容。

开发步骤：

① 安装配置环境

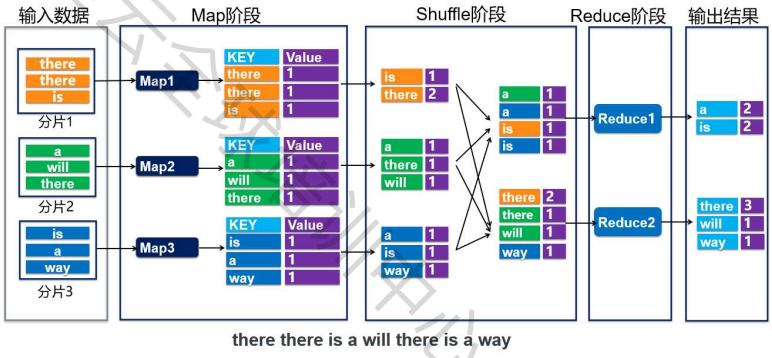
② 开发 MR 程序

③ 本地模式测试脚本

④ 导出 Jar 包

⑤ 上传至 MaxCompute 项目空间

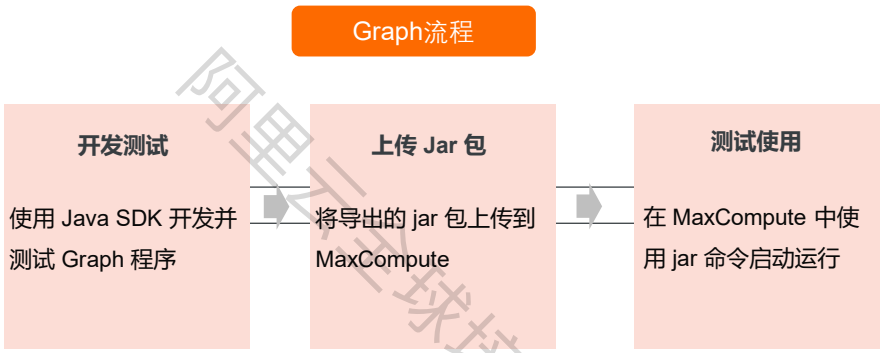
⑥ 在 MaxCompute 中使用MR



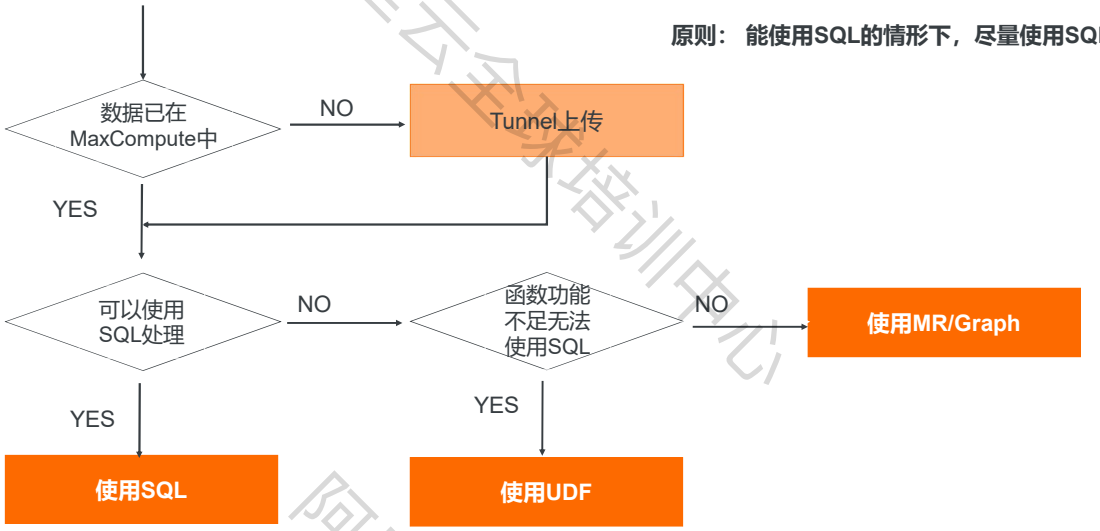
数据开发工具-Graph

- **MaxCompute Graph** 是一套面向迭代的图计算处理框架。
  - 图计算作业使用图进行建模，图由点(Vertex)和边(Edge)组成，点和边包含权值(Value)。
  - 通过迭代对图进行编辑、演化，最终求解出结果
  - 典型应用：PageRank，单源最短距离算法 ，K-均值聚类算法 等等。
  - 用户可以使用 MaxCompute GRAPH 提供的接口Java SDK编写图计算程序。
  - 图计算用于挖掘人、物等实体之间的潜在的不易观察的行为和联系，这些联系很难通过关系型数据库表示。

Graph 开发使用流程



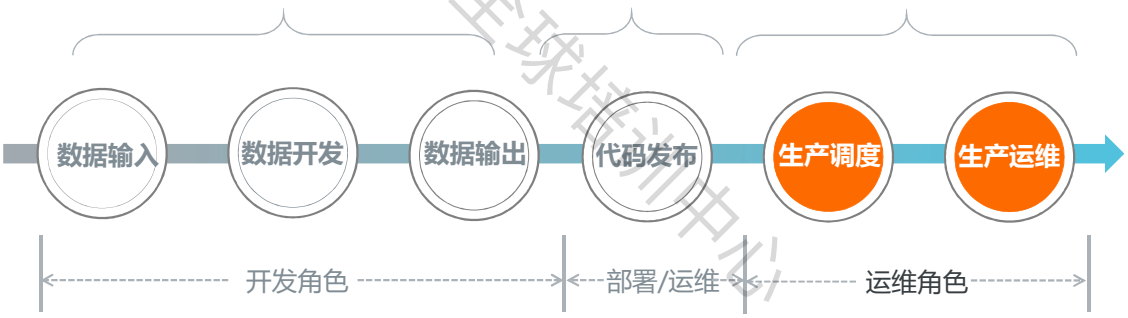
如何选择合适的使用方式



课程目录

应用流程

涉及模块：数据集成模块　数据开发模块　发布管理模块　运维中心模块

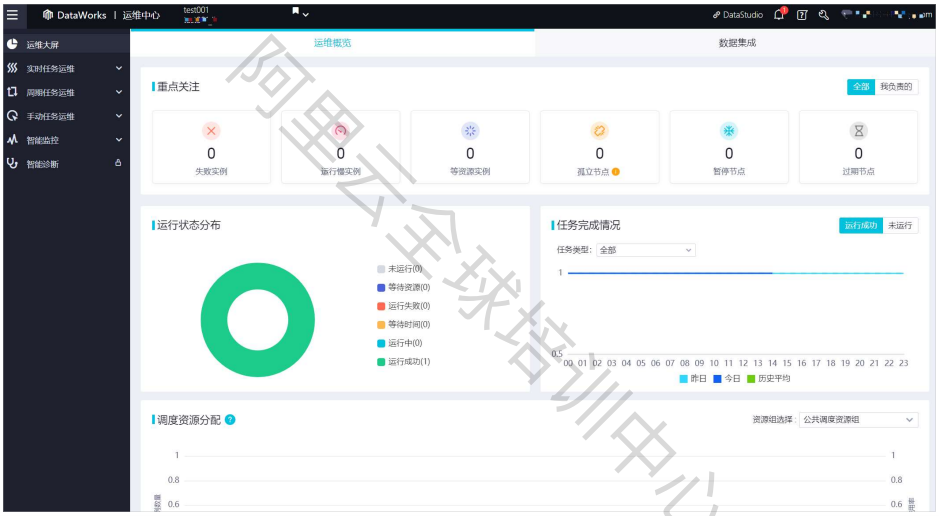


数据开发完成的任务需要部署到生产调度系统，因此调度系统每天需要运行大量数据处理任务，任务之间依赖复杂，为保障任务按时正常运行，保障系统正常运转所做的相关操作都属于运维。DataWorks运维中心为运维人员提供了便捷、智能的任务监控和任务运维工具，为运维工作提供了有力支撑。

阿里云

任务运维-运维中心

运维中心包括运维大屏、实时任务运维、周期任务运维、手动任务运维、智能监控和智能诊断



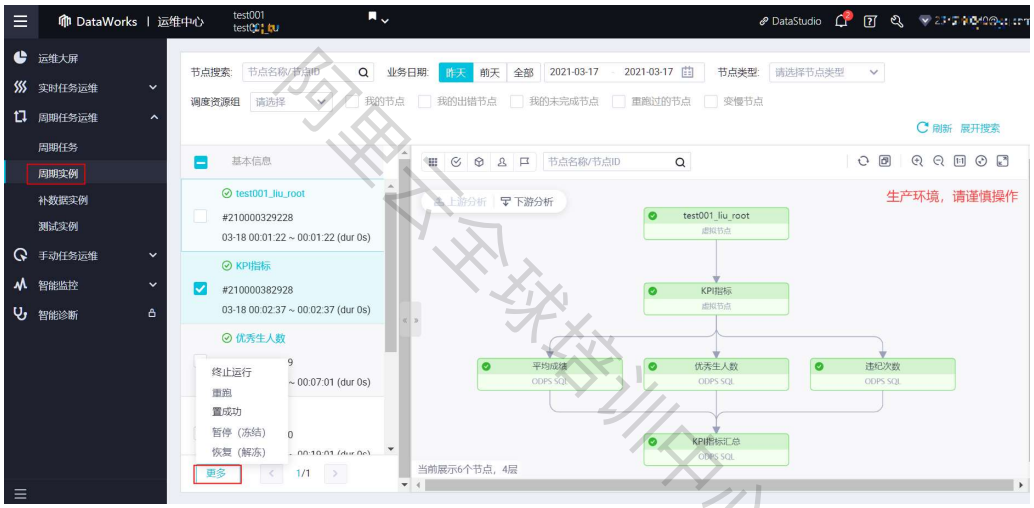
阿里云

任务运维-任务（周期、手动）可进行的操作

	手动任务	周期任务
运行方式	手动触发	系统调度
任务筛选条件	类型、节点名称、节点类型、责任人、今日修改的节点等	节点名称、解决方案、业务流程、节点类型、责任人、基线、今日修改的节点等
操作	运行、查看实例、修改责任人和资源组、查看血缘	测试、补数据、冻结、解冻、查看实例、添加报警、修改责任人和资源组、配置告警、质量监控等
批量操作	批量修改责任人	批量进行添加报警、修改责任人、修改资源组、冻结、解冻操作和添加到基线和下线节点等操作。
节点代码	可查看不能修改	

任务运维-任务实例运维

对任务实例可进行的操作：终止运行、重跑、置成功、暂停（冻结）、恢复（解冻）



任务运维-智能监控

智能监控模块是DataWorks（数据工场）任务运行的监控及分析系统。根据监控规则和任务运行情况，智能监控决策是否报警、何时报警、如何报警以及给谁报警。智能监控会自动选择最合理的报警时间，报警方式以及报警对象。

基线预警和事件告警	自定义提醒
• 通过设定基线监控任务，即监控范围	• 轻量级监控功能
• 设定报警策略	• 自行设定报警对象、条件、方式以及频次
• 智能判定报警时机和对象、自动报警	• 触发条件包括完成、出错、未完成、超时

智能监控-基线预警与事件告警示例

要满足基线A的基线时间5:00，则要求：任务B的最晚完成时间：4:10

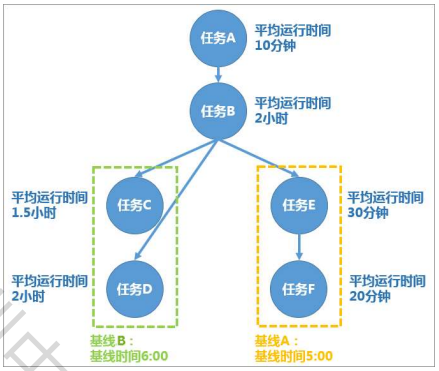
要满足基线B的基线时间6:00，则要求：任务B的最晚完成时间：4:00

时间倒推

基线A、B同时满足：任务B的最晚完成时间：4:00

任务A的最晚完成时间：2:00

任务A的最晚开始时间：1:50



假设1点时任务A运行出错，则此时任务A的**余量**时间即为1:50和1:00之间的差值，即50分钟。

基线任务配置预警余量和承诺时间，当智能监控预测基线的完成时间超过预警**余量**时触发告警（目前通知3次，间隔30分钟）。

课程目标

学习完本课程后，你将能够：

1. 了解数据质量基本概念、重要性、质量评价方式
2. 熟悉数据质量分析方法，掌握业务、技术、管理等几个维度
3. 掌握数据质量管理的方法论体系
4. 掌握数据质量事前、事中、事后控制策略及控制流程

课程目录

数据质量概念

- 数据是对现实世界的反映，数据质量指的是数据在多大程度上反映了真实世界。一般来说，如果数据代表的意义与目的不一致，数据就有质量问题。

现实世界	数据管理系统（DMS）
在现实世界中的行为和事件产生的结果，有些能够被数据化	通过收集现实世界中被数据化的行为和事件，以结构化的形式将其映射到一个数据库

数据质量: DMS多大程度上反映了真实的世界？0% - 100%?

现实世界 ? DMS

数据质量的重要性

- 如果数据质量存在问题，即数据无法正确的反映真实世界的状况，模型就会出现较大偏差，从而预测出有偏差甚至完全相反的结果。
- **据IBM研究统计：**
  - 错误或不完整数据导致BI和CRM系统不能正常发挥优势甚至失效。
  - 数据分析员每天有30%的时间浪费在了辨别数据是否是“坏数据”上。
  - 低劣的数据质量严重降低了全球企业的年收入。

糟糕的数据质量常常意味着糟糕的业务决策，

将直接导致数据统计分析不准确、监管业务难、高层领导难以决策等问题

数据质量评价维度

- 数据质量 (DQ)维度是数据管理从业者基于已有的一些标准用来评估和衡量数据质量的一些特征。国际数据管理协会（DAMA）对数据质量的衡量标准，分为以下六大维度。

- 衡量所必须的数据的完整程度。
- 针对用户对信息获取的时间及时性要求，确保数据及时更新。
- 针对某个数据项或某组数据，没有重复的数据值
- 对于数据的值、格式要求符合数据定义或业务定义的要求。
- 数据元素的类型和含义必须一致和清晰。
- 确保数据必须反映真实的业务内容。

数据质量评价维度-完整性

- 数据完整性问题包括：模型设计不完整，例如：唯一性约束不完整、参照不完整；数据条目不完整，例如：数据记录丢失或不可用；数据属性不完整，例如：数据属性空值。不完整的数据所能借鉴的价值就会大大降低，也是数据质量问题最为基础和常见的一类问题。

客户ID	名	姓	城ID	省ID	电话	地址	城市	邮编	收入	婚姻
1	小二	阮	44	131	360-9430326	江城路	杭州	310014	60K	null
2	小五	袁	44	131	615-7581118	建国路	杭州	310014	96K	结婚
3	小七	张	01	908	252-2230627	之江路	南京	310001	90K	结婚
4	圆圆	王	01	908	781-7778794	之江路	南京	310001	50K	单身
5	圆圆	吴	01	908	401-5351127	之江路	南京	310001	50K	结婚
6	圆圆	李	04	131	318-8835754	江城路	大连	280012	80K	结婚

元组 1 存在属性值缺失（婚姻），是否存在元组缺失需要根据更多信息判断

数据质量评价维度-唯一性

- 用于识别和度量重复数据、冗余数据。重复数据是导致业务无法协同、流程无法追溯的重要因素，也是数据治理需要解决的最基本的数据问题。如下：

客户ID	名	姓	城ID	省ID	电话	地址	城市	邮编	收入	婚姻
1	小二	阮	44	131	360-9430326	江城路	杭州	310014	60K	单身
2	小五	袁	44	131	615-7581118	建国路	杭州	310014	96K	结婚
3	小七	张	01	908	252-2230627	之江路	南京	310001	90K	结婚
4	圆圆	王	01	908	781-7778794	之江路	南京	310001	50K	单身
5	圆圆	王	01	908	781-7778794	之江路	南京	310001	50K	单身
6	圆圆	李	04	131	318-8835754	江城路	大连	280012	80K	结婚

元组 4、5 属于重复数据

数据质量评价维度-及时性

- 数据的及时性(In-time)是指能否在需要的时候获到数据，数据的及时性与企业的数据处理速度及效率有直接的关系，是影响业务处理和管理效率的关键指标。

客户ID	名	姓	城ID	省ID	电话	地址	城市	邮编	收入	婚姻
1	小二	阮	44	131	360-9430326	江城路	杭州	310014	60K	单身
2	小五	袁	44	131	615-7581118	建国路	杭州	310014	96K	结婚
3	小七	张	01	908	252-2230627	之江路	南京	310001	90K	结婚
4	圆圆	吴	01	908	318-8835754	之江路	南京	310001	50K	单身
5	圆圆	李	01	908	318-8835754	之江路	南京	310001	50K	结婚
6	圆圆	李	04	131	318-8835754	江城路	大连	280012	80K	结婚

元组 4、5、6可能描述的是同一个人的信息，每一条的时效性?

数据质量评价维度-有效性

- 数据有效性数据只有服从业务属性(格式,类型,范围)才是有效的。对于数据的值、格式要求符合数据定义或业务定义的要求，如某些电话、邮箱的格式。

客户ID	名	姓	城ID	省ID	电话	地址	城市	邮编	收入	婚姻
1	小二	阮	44	131	360-9430326	江城路	杭州	310014	60K	结婚
2	小五	袁	44	131	615-7581118	建国路	杭州	310014	96K	结婚
3	小七	张	01	908	252-2230627	之江路	南京	310001	90K	结婚
4	圆圆	王	01	908	781-777879X	之江路	南京	310001	50K	单身
5	圆圆	吴	01	908	401-5351127	之江路	南京	310001	50K	结婚
6	圆圆	李	04	131	318-8835754	江城路	大连	280012	80K	结婚

元组 4的电话数据格子异常、为无效号码

数据质量评价维度-准确性

- 数据准确性也叫可靠性，是用于分析和识别哪些是不准确的或无效的数据，不可靠的数据可能会导致严重的问题，会造成有缺陷的方法和糟糕的决策。如下：

客户ID	名	姓	城ID	省ID	电话	地址	城市	邮编	收入	婚姻
1	小二	阮	44	131	360-9430326	江城路	杭州	310014	60K	单身
2	小五	袁	44	131	517-7758716	建国路	杭州	310014	96K	结婚
3	小七	张	01	908	252-2230627	之江路	南京	310001	90K	结婚
4	圆圆	王	01	908	781-7778794	之江路	南京	310001	50K	单身
5	圆圆	吴	01	908	401-5351127	之江路	南京	310001	50K	结婚
6	圆圆	李	04	131	318-8835754	江城路	大连	280012	-80K	结婚

元组 6的收入可能是不准确的

数据质量评价维度-一致性

- 多源数据的数据模型不一致，例如：命名不一致、数据结构不一致、约束规则不一致。数据实体不一致，例如：数据编码不一致、命名及含义不一致、分类层次不一致、生命周期不一致等。相同的数据有多个副本的情况下的数据不一致、数据内容冲突的问题。

客户ID	名	姓	城ID	省ID	电话	地址	城市	邮编	性别	婚姻
1	小二	阮	44	131	360-9430326	江城路	杭州	310014	M	单身
5	圆圆	吴	01	908	401-5351127	之江路	南京	310001	F	结婚
6	圆圆	李	04	131	318-8835754	江城路	大连	280012	F	结婚

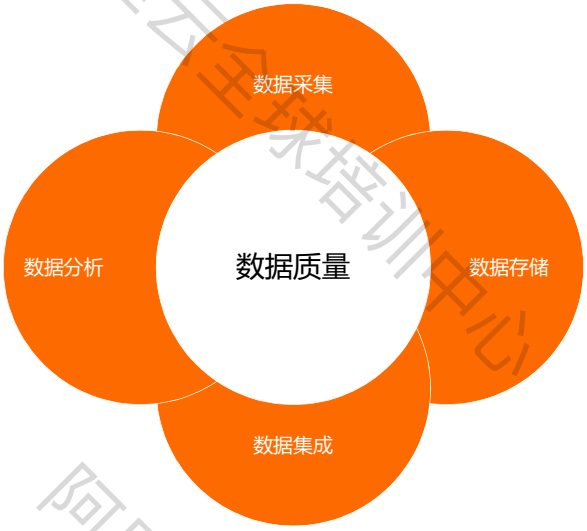
客户ID	名	姓	城ID	省ID	电话	地址	城市	邮编	性别	婚姻
1	小二	阮	44	131	360-9430326	江城路	杭州	310014	Man	单身
5	圆圆	吴	01	908	401-5351127	之江路	南京	310001	Female	结婚
6	圆圆	李	04	131	318-8835754	江城路	大连	280012	Female	结婚

由于数据源不同，对于一组数据，分别用M和F，Male和Female表示性别，产生编码不一致性的问题。

课程目录

1. 数据质量基本概念
2. **数据质量问题的来源**
3. 数据质量问题的控制
4. 数据质量控制最佳实践

数据质量问题的来源-技术方面



数据采集问题

- 采集标准缺失，或标准执行不到位
  - 例如：注册信息表没有进行数据格式约束，或填写时出现错拼、漏填等现象
  - 问题：规范性，准确性，完整性
- 源数据问题
  - 例如：爬取得到的网页数据，下载或购买得到的数据集
  - 问题：完整性，准确性，时效性
- 软件、硬件、网络问题
  - 例如：机器传感器故障，流计算时网络延迟
  - 问题：完整性，时效性

调查问卷表

BM	XB	NL	SG	SR
30001	女	36.	5’ 3”	30000
2987	男	27	175	20k
9527		49	5’ 4”	8000
101	女	16	5’ 7”	

投融资数据

时间	公司名称	行业	轮次	金额	投资方
2019/3/11	儿童周末	教育	Pre-A轮	数百万人民币	北塔资本
2019/3/11	SOM Biotech	医疗健康	A轮	700万欧元	未透露
2019/3/11	Klaytn	企业服务	战略投资	9000万美元	未透露
2019/3/11	谊品生鲜	本地生活	B轮	未透露	腾讯
2019/3/11	常仁科技	企业服务	A轮	1亿人民币	未透露

识别一下两个表格中的数据质量问题

数据存储问题

- 元数据管理水平低下
  - 例如：血缘关系不明（表之间、属性之间），属性的含义不明确
  - 问题：准确性
- 漏洞百出的数据模型
  - 例如：失踪的主键、时间戳等
  - 问题：唯一性，时效性
- 软件、硬件、网络问题
  - 例如：千禧年问题，灾害，双11活动等
  - 问题：准确性，完整性

customers

ID	GENDER	AGE	HEIGHT
1	M	36	176
2	F	27	162
3	F	49	174
4	M	16	165

kh

BM	XB	HF	SG
30001	女	YH	5’ 3”
2987	男	YH	5’ 9”
9527	男	WH	5’ 4”
101	女	YH	5’ 7”

两个表的关系未知、BM和HF属性含义未知

数据集成问题

- 源数据性质不同或者处理不当
  - 例如：同义不同名、同名不同义、数据截断/删失等
  - 问题：规范性、准确性
- 业务流程中的装载规则设置错误
  - 例如：覆盖（Insert Overwrite） vs 新增（Insert Into）
  - 问题：唯一性，完整性
- 社会学因素
  - 例如：企业内部各集团或部门之间数据孤岛
  - 问题：完整性

customers

ID	GENDER	AGE	HEIGHT
1	M	36	176
2	F	27	162
3	F	49	174
4	M	16	165

kh

BM	XB	HF	SG
30001	女	YH	5’ 3”
2987	男	YH	5’ 9”
9527	男	WH	5’ 4”
101	女	YH	5’ 7”

性别、身高属性同义不同名

数据分析问题

- 没有及时探查数据
  - 例如：Join时，没有发现笛卡尔积，聚合后得到错误结果
  - 问题：唯一性，准确性
- 处理手法有误
  - 例如：没有使用合理的抽样方法，造成分析结果偏差
  - 问题：完整性，准确性

sales_id	prod_id	quantity
100014	934	1
100015	935	9
100016	936	7
100017	937	2

prod_id	prod_name
934	b1994
935	B1995
936	B1996
937	B1997
937	b1997

sales_id	prod_id	quantity	prod_name
100014	934	1	b1994
100015	935	9	B1995
100016	936	7	B1996
100017	937	2	B1997
100017	937	2	b1997

prod_id	prod_name	total_q
934	b1994	1
935	B1995	9
936	B1996	7
937	B1997	4

```
SELECT prod_id, prod_name, SUM(quantity) as total_q
FROM t1 a LEFT JOIN t2 b ON a.prod_id = b.prod_id
GROUP BY prod_id
```


数据质量问题来源-其它方面

- **业务方面主要体现在**
  - 业务需求不明确
  - 需求频繁变更
  - 数据输入不规范
  - 业务数据做假

规范化的业务流程

是保障数据质量管理有条不絮的进行

- **管理方面主要体现在**
  - 缺乏数据思维
  - 缺乏认责机制
  - 缺乏明确目标
  - 缺乏统一管理机制
  - 缺乏统一数据规范
  - 缺乏质量控制措施
  - 缺乏问题管理制度

完善的管理机制

是保障是数据质量控制的关键

课程目录

数据质量控制策略

- 数据质量控制策略应该形成数据质量的**事前预防控制、事中过程控制和事后监督控制**，进行数据质量闭环管理。

数据质量控制策略-事前预防控制

- 数据质量问题的事前预防控制最有效的方法就是找出发生数据质量问题的根本原因并采取相关的策略进行解决。

- **主要方法**
  - 确定根本原因
  - 实施改进方案
- **主要手段**
  - 建立数据标准化模型
  - 构建数据分类和编码体系

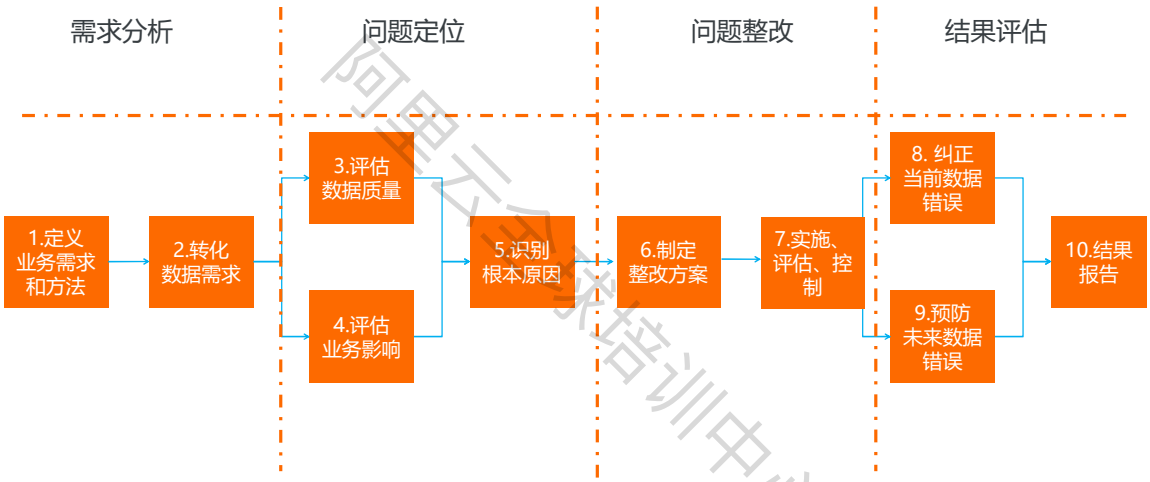
数据质量控制策略-事中过程控制

- 事中数据质量的控制，即在数据的维护和使用过程中去监控和处理数据质量。通过建立数据质量的流程化控制体系，对数据的新建、变更、采集、加工、装载、应用等各个环节进行流程化控制。

数据质量控制策略-事后监督控制

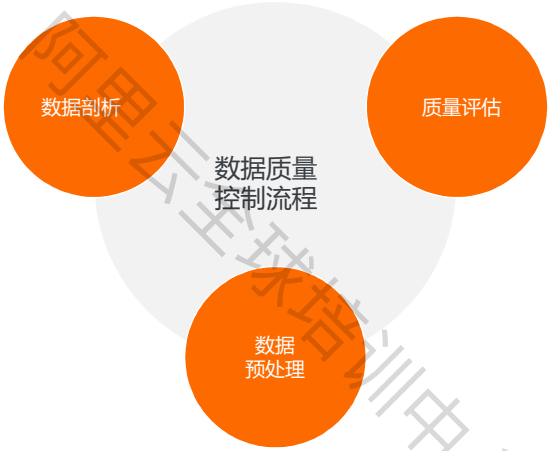
- 数据质量的事后监督控制尤为重要，定期开展数据质量的检查和清洗工作应作为企业数据质量治理的常态工作来处理，主要通过以下步骤和策略：

解决和改进质量问题的主要步骤



课程目录

数据质量控制流程



数据剖析

- 定义

  对数据内容进行系统化的**探查、统计和记录。**

- 内容

  结构剖析：表和表之间、列和列之间的依赖关系，数据粒度，数据精度，业务规则等。

  列属性剖析：列类型，列长度，重复记录，列值域，列默认值，列空值，列有效性等。

- 作用

  描述了数据的结构、内容、规则和关系的概况，获得对数据和环境的了解，是变被动为主动地改善数据质量的第一步。

数据剖析-原因探查

- 完整性

  探查每一个列是否有空值(Null)或空字符串（’ ’）；探查维表内是否包含事实表所有主键。

- 唯一性

  检查单列是否有重复值；检查某一个列的组合是否有重复值。

- 及时性

  检查是否有时间戳，时间戳是否满足其他四个维度；检查某一个时间段内是否没有记录。

- 有效性

  检查数据的业务属性(格式,类型,范围）；数据类型(字符串,整数型,浮点型,指针型等), 格式(长度,数量等)和范围(最小,最大,或在允许的数值范围内)。

- 准确性

  根据业务规则，进行约束检查；根据统计信息，判断是否异常。

- 一致性

  对比两个或多个业务定义规则或口径差异；检查跨多数据集的一致性分析或跨多数据项的数据值和格式。

数据剖析-统计记录

数据库表	字段	空值个数	非空个数	总数	类数	最大值	最小值	均值
Ecom_fact_sales	Product_key	0	61512	61512	13828	1002557	102	-
	Date_key	0	61512	61512	30	42642	42613	-
	Line_item_total_amount	0	61512	61512	-	5000	0.01	57.2012
	Customer_key	0	61512	61512	51737	410000	8	-
	Website_key	0	61512	61512	13828	98759	1	-
Ecom_dim_product	Product_key	0	38213	38213	38213	1002617	100	-
	department	0	38213	38213	19	Games	appliances	-
Ecom_dim_customer	Customer_key	0	410276	410276	410276	410277	1	-
Ecom_dim_date	Date_key	0	47847	47847	47847	47847	1	-
	Calendar_date	0	47847	47847	47847	2030/12/31	1900/1/1	-

其它检查项目	其它项目结果
事实表的Product_key都在维表内	是
事实表的Customer_key都在维表内	是
事实表的date_key都在维表内	是

数据质量控制流程-质量评估

- 一个合格的数据分析师在评估数据质量时，除了统计数据质量的6维度问题外，还需要结合具体的业务情况来进行综合考量，以得到正确的处理方式。
  - 通常情况下，异常值是数据质量问题，但有时候异常值可能是合理合法的
    - 例如：高利润的客户（数据矿工的梦想 ）
    - 例如：机器运行故障（想要监控的情报 ）
  - 数据质量问题的评估需要行业的知识，对数据的正确理解和行业专家的配合
    - 行业知识和数据理解很少有书面记录，通常存储在人们的头脑中
    - 行业知识和数据理解通常分散在企业组织的各个部门
    - 往往各个专家会意见不统一，因此需要调查研究并总结

数据质量控制流程-质量评估

- 考虑下面这个有完整性问题的客户维度表，在不同业务场景下，会如何评估?

  假设业务部门需要了解在2016年，单身人群与结婚人群的消费金额对比

  - 缺失数据量不大的时候，通常不影响客户画像类型的分析，或影响轻微。

  假设业务部门需要为单身人群量身定做一套促销方案

  - 缺失数据导致促销方案中的有效客户数量下降，对促销有一定影响。

  假设业务部门需要上线一套客户个性化推荐的服务，其中婚姻状况是主要考量因素

  - 缺失数据导致部分顾客无法享受该服务，对业务有绝对的影响。

名	姓	城ID	省ID	地址	城市	2016年消费	邮编	收入	婚姻
小二	阮	44	131	江城路	杭州	￥2000	310014	60K	单身
小七	张	01	908	之江路	杭州	￥3000	310001	90K	NULL
成功	李	01	908	之江路	南京	￥5000	310001	50K	NULL
建国	赵	04	131	江城路	大连	￥4000	280012	80K	结婚

数据质量控制流程-数据预处理

数据清洗

Data Cleaning

- 完整性、唯一性
- 及时性、有效性
- 准确性、一致性

- 类型转换
- 去重
- 处理异常值
- 缺失值填充

数据加工

Data Processing

- 采样与过滤
- 行列操作
- 归一化
- 标准化
- 特征工程

- 随机、加权、分层采样，过滤与映射
- JOIN，UNION，序号列，拆分
- 特征变换，特种重要性评估，特征选择，特征生成

数据质量控制流程-数据清洗

规范性问题

- 常用方法：定义标准元数据——定义转换规则（创建Map表）——转换数据。

customers

1	M	36	176
2	F	27	162
3	F	49	174
4	M	16	165

map_gender

M	1
F	2
女	2
男	1

ods_customers

1	1	36	176
2	2	27	162
3	2	49	174
4	1	16	165

kh

BM	XB	HF	SG
30001	女	YH	5’ 3”
2987	男	YH	5’ 9”
9527	男	WH	5’ 4”
101	女	YH	5’ 7”

map height
公式
round(x*12+y)*2.54,0)

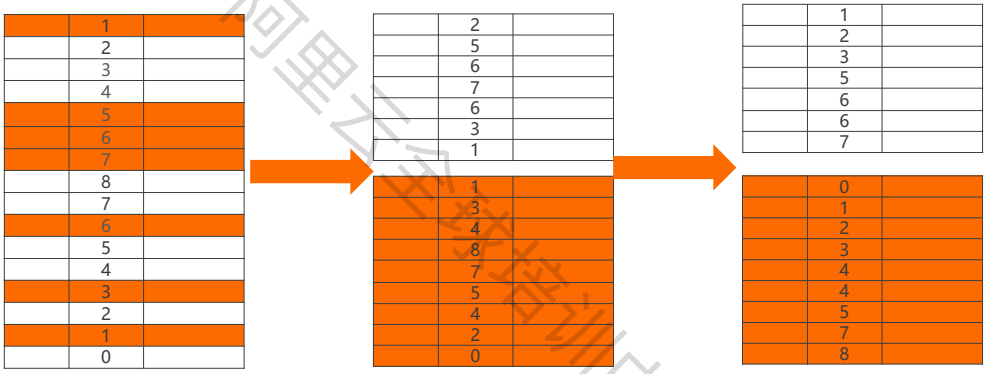
ods_kh

BM	XB	HF	SG
30001	2	YH	160
2987	1	YH	175
9527	1	WH	163
101	2	YH	170

数据质量控制流程-数据清洗

唯一性问题

- 常用方法：使用distinct 或者group by 保留一条记录；使用开窗函数，保留指定的记录。



原始数据　　Partition　　Order by

数据质量控制流程-数据清洗

准确性问题

- 常用方法：删除异常值所在元组，或者用业务规则制定的极值代替。
- 例：假设已知价格的上限为1000元 。

Book_key	Book_title	author	asin	publisher	Publication_date	pages	price
75	Database Technologie	Join	97816056605	IGI Global	2009-02-28 00:00:00	2962	1650
495	Encyclopedia data	Wei Wang	97816056601	IGI Global	2008-08-28 00:00:00	2550	1195
803	Data waerhousing	Wang	97815990495	IGI Global	2008-05-28 00:00:00	4104	1841.58

A	B	C	D	E	F	G	H
Book_key	Book_title	author	asin	publisher	Publication_date	pages	price
75	Database Technologie	Join	97816056605	IGI Global	2009-02-28 00:00:00	2962	1000
495	Encyclopedia data	Wei Wang	97816056601	IGI Global	2008-08-28 00:00:00	2550	1000
803	Data waerhousing	Wang	97815990495	IGI Global	2008-05-28 00:00:00	4104	1000

数据质量控制流程-数据清洗

完整性问题

- 常用方法：使用最可能的值填充缺失值，包括：
  - 使用一个全局常量（NULL统一由-1代替，代表未知）
  - 使用统计值（均值/极值/众数等）
  - 使用拟合值（通过模型或者规则拟合，得到“最应该”填的值 ）

ods_customers

ID	GENDER	AGE	HEIGHT
1	1	36	176
2	2	27	NULL
3	2	NULL	164
4	1	16	165
5	1	23	180
6	NULL	31	159
...	...	...	...

ods_customers

ID	GENDER	AGE	HEIGHT
1	1	36	176
2	2	27	162
3	2	27	164
4	1	16	165
5	1	23	180
6	-1	31	159
...	...	...	...

数据质量控制流程-数据清洗

时效性问题

- 常用方法：结合冗余记录和时效约束来恢复时序关系。

tID	EID	FN	LN	City	Salary	Status	Time
t1	1	Alice	Smith	Beijing	50k	Single	T1
t2	1	Alice	Smith	Shanghai	70k	Single	T2
t3	1	Alice	Green	Guangzhou	80k	Married	?
t4	1	Alice	Green	Harbin	80k	Married	?

假设该数据集中无可用时间戳可以表示数据的生成时间及插入时间，但有时效约束如下：

- cc1: 同一个人的工资不会随时间下降。
- cc2: 存有工资最新值的记录也存有工作城市的最新值。
- cc3: 同一个人的婚姻状况只能由 Single 变成 Married，再变成 Divorced。

可以根据推理，得出部分时序关系。

数据质量的长期保持

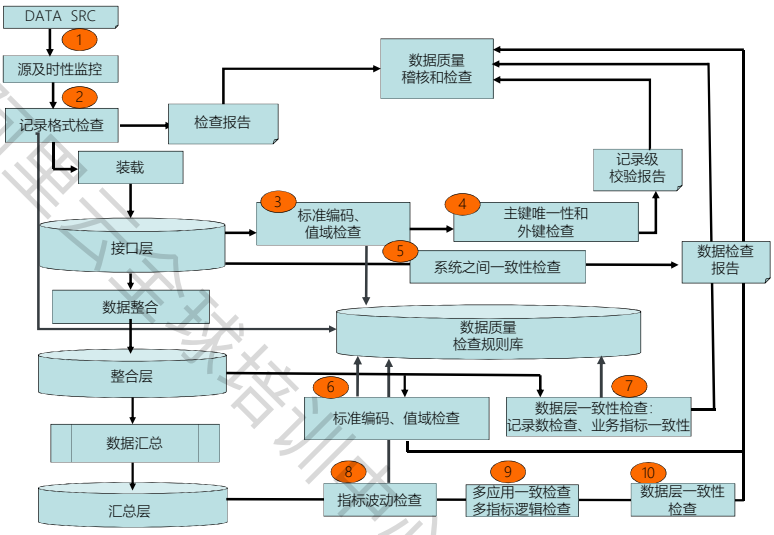
- 数据质量的保证**仅靠技术显然是不行的。**在整个数据质量的控制过程中，**人是关键因素。**要想真正长期保证数据的高质量，还必须从以下5个方面着手：
  - 建立数据的标准，明确数据的定义（权衡企业内外部因素，即习惯和质量）。
  - 建立一个可复用的数据收集、数据预处理和数据维护流程（应对不断变化的企业内外部因素）。
  - 在数据预处理流程中设立多个性能监控点（评价标准：最终用户、同类数据、前期数据等）。
  - 对流程不断进行改善和优化（质量改善非一朝一夕，而是持续过程，要灵活变通）。
  - 把责任落实到人（制定数据采集、存储、集成、分析等各项活动的质量指标）。

课程目录

1. 数据质量基本概念
2. 数据质量问题的来源
3. 数据质量问题的控制
4. **数据质量控制最佳实践**

数据质量分析

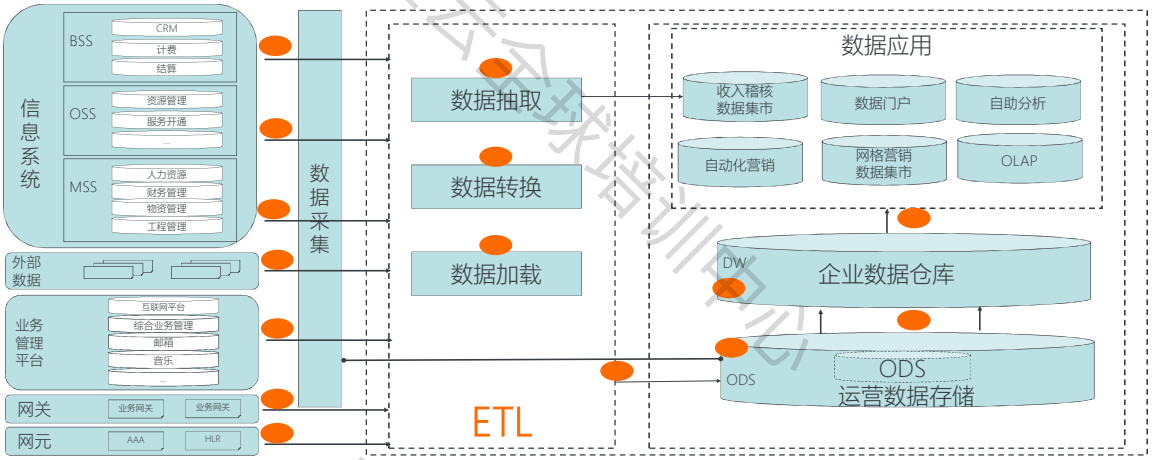
- **数据源检查**
  - 1、源及时性监控
  - 2、记录格式检查
- **接口层检查**
  - 3、标准编码、值域检查
  - 4、主键唯一性和外键检查
  - 5、系统之间一致性检查
- **整合层检查**
  - 6、标准编码、值域检查
  - 7、数据层一致性检查：
    记录数检查、业务指标一致性检查
- **汇总层检查**
  - 8、指标波动检查
  - 9、多应用一致检查、多指标逻辑检查
  - 10、数据层一致性检查



阿里云全球培训中心

阿里云

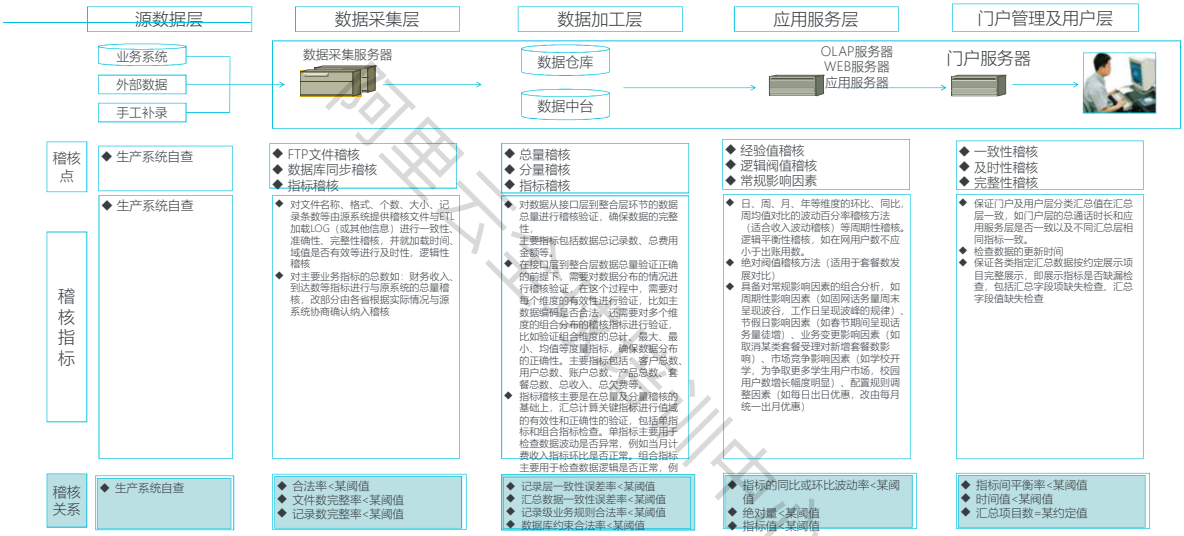
管理稽核点



管理稽核点

贯穿数据采集到应用端到端的数据质量管理

稽核点及稽核指标



阿里云

数据分析指标

阿里云大数据分析师ACP认证课程

课程目标

学习完本课程后，你将能够：

1. 掌握数据指标的概念、分类、衡量标准
2. 理解数据指标构建方法

课程目录

1. 数据分析指标基础

2. 数据分析指标体系

指标的概念

- **指标**（*Indicator*）：指预期中打算达到的指数、规格、标准
- 完整的统计指标（Statistical Indicator）
  - ✓ 指标名称，说明所反映现象数量特征的性质和内容
  - ✓ 统计的时间界限和空间范围
  - ✓ 计量单位
  - ✓ 指标的数值

指　　标	
运营	销售额
	毛利率
客户价值	客单价
	消费频率
商品类目	SKU数
	SPU数

例：中国1983年全国钢产量4002万吨

指标的衡量标准

- 好的数据指标是**一个比率**
  - 比率的可操作性强，是行动的向导
  - 比率是天生的比较性指标
  - 比率还适用于比较各种因素间的相生和相克（正相关和负相关）

可比较	可理解	比率
• 不同的时间段 • 不同用户群体 • 不同竞争产品	• 简单易懂 • 避免误导 • 业务解释性	• 可操作性强 • 可以比较 • 反映关联性

指标的分类

定性和定量

定性指标 非结构化的、经验性的、揭示性的、难以归类的指标

定量指标 可以被计数和衡量的指标

【例】在绩效考核当中

- 管理层：适宜采用定量成分较多、约束力较强、独立性较高、以最终结果为导向的考核指标
- 普通员工：适宜采用定性成分多、需要上下级随时充分沟通、主要以工作过程为导向的考核指标

虚荣和北极星

虚荣指标 容易给人留下印象，但是无法用于决策的指标

北极星指标 可以指明工作的方向，帮助改进商业模式，决策下一步行动的指标

【例】

- 虚荣指标：总注册用户数
- 北极星指标： 总活跃用户数 、总活跃用户占比

先见性和后见性

先见性指标 用于预测企业未来情况的指标
通过对未来的预测，制定利益最大化的策略

后见性指标 用于揭示当前存在问题的指标
发现存在的问题后通过干预，减少损失

【例】

- 先见性指标：下半年每个月的预测销量
- 后见性指标：用户流失率变化

相关和因果

相关指标

一个或多个指标变化的同时，另一个或多个指标也会随之变化
但是指标之间不存在先后的时间关系

因果指标

一个或多个指标的改变，能够对另一个或多个指标产生某种作用
指标之间的变化存在先后的时间关系

【例】

- 相关指标：雪糕销量和溺亡人数
- 因果指标：商品单价和销售额

指标常见分析方法

- 对比分析
- 分组分析法
- 结构分析法
- 平均和变异分析法
- 交叉分析法

课程目录

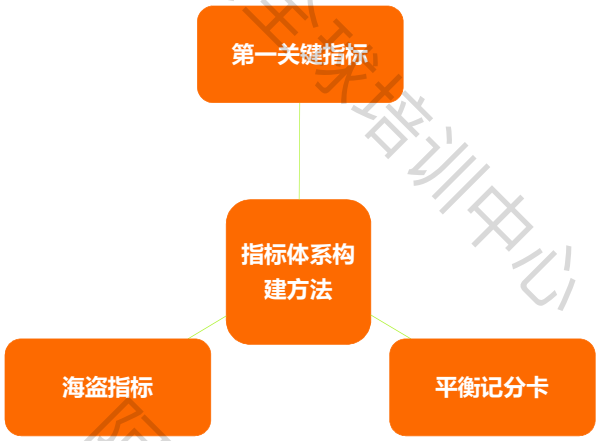
1. 数据分析指标基础
2. **数据分析指标体系**

什么是指标体系

- **指标体系**（*Indicator System*）：在业务环境中，由一系列相互之间有逻辑联系的指标所组成的整体，从各个侧面反映出现象总体或样本的数量特征。
- **构建原则**：系统性、典型性、动态性

指标体系的构建方法

- 常见的构建方法有三种，不同的方法适用于不同的场景。



基于第一关键指标

- **第一关键指标**（***O**ne **M**etric **T**hat **M**atters*，***OMTM***）是***精益数据分析***中提出的概念，其主要含义：
  - 企业在任何一个阶段都需要找到这样的一个数字，将这个数字放到比其他任何事情都要重要的位置上，因此需要在众多的指标中筛选出**影响最大**的指标
  - 小企业比较容易设置，业务复杂的中大型企业，有很多完全不同的业务部门，很难找到 OMTM

- 第一关键指标的作用有：
  - **回答了现阶段最重要的问题**：随时间而变化，不同阶段关注不同问题（MVP- 增长）
  - **促使建立清晰的目标**：定义什么是成功，设定好目标
  - **关注整个公司层面的健康**：没人喜欢“数据呕吐”，把OMTM放到所有地方
  - **鼓励一种实验文化**：经历小的挫败，能避开大的失败

值得一提：

一些孵化项目的投资人考量创业团队的重要依据之一就是团队能否足够清晰地理解并跟踪自己的 OMTM

第一关键指标体系构建方法

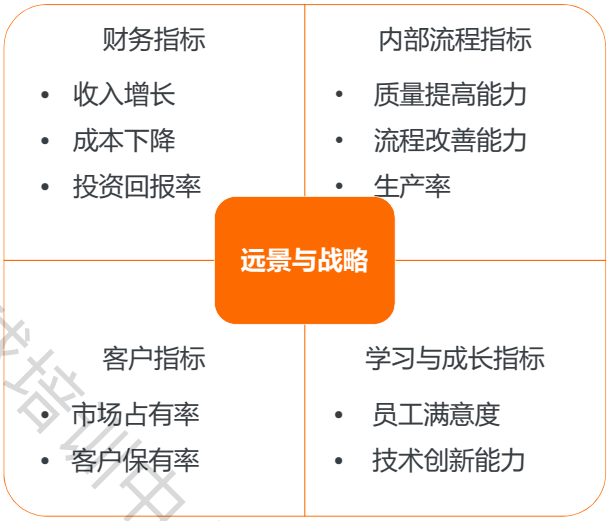
- 步骤如下：
  1. 选择第一关键指标
  2. 寻找与第一关键指标相关的因素
  3. 拆解第一关键指标（MECE原则）

- 第一关键指标的选择非常重要，需要注意以下问题：
  - ✓ 该指标能否衡量客户是否体验到了产品的核心价值?
  - ✓ 该指标能否反映客户的活跃度?
  - ✓ 该指标的变化能否反映出公司的状况?
  - ✓ 该指标是否容易被团队理解?
  - ✓ 该指标是否存在严重的滞后性?
  - ✓ 该指标是否存在可操作性?

基于平衡记分卡

- **平衡记分卡**：以企业战略为导向，通过财务、客户、内部业务流程和学习与成才四个方面及其业绩指标的因果关系，全面管理和评价企业综合业绩，是企业愿景和战略的具体体现，既是一个绩效评价系统也是一个有效的战略管理系统。

- 平衡记分卡的主要特点：
  - ✓ 自上而下地体现着公司的使命和战略
  - ✓ 具有前瞻性
  - ✓ 整合了外部和内部的衡量指标
  - ✓ 帮助聚焦重点事项



基于海盗指标

- **海盗指标：**以**收入和增长**为目标，通过内在的 5 个指标之间逻辑关系，改善并推动业务增长

要素	目标	相关指标
获取 Acquisition	通过各种方式吸引顾客	流量，提及量，搜索成本，获客成本
激活 Activation	将获取的用户转换为产品真正的使用者	注册用户数，注册量，新手教程完成量，订阅量
留存 Retention	说服用户再次光临，反复使用，表现出粘性行为	用户参与度，（日/月）活跃用户数，流失率
推荐 Referral	已有用户对潜在用户的传播	邀请发送量，病毒式传播，病毒传播周期
营收 Revenue	商业活动的产出	客户价值，广告营收，付费转换率

讨论： 哪些企业适合使用海盗指标构建指标体系？

阿里云

数据编程规范

阿里云大数据分析师ACP认证课程

阿里云

课程目标

学习完本课程后，你将能够：

1. 掌握数据编程过程及数据分析用表处理方法
2. 了解数据编程规范、编程效率

阿里云

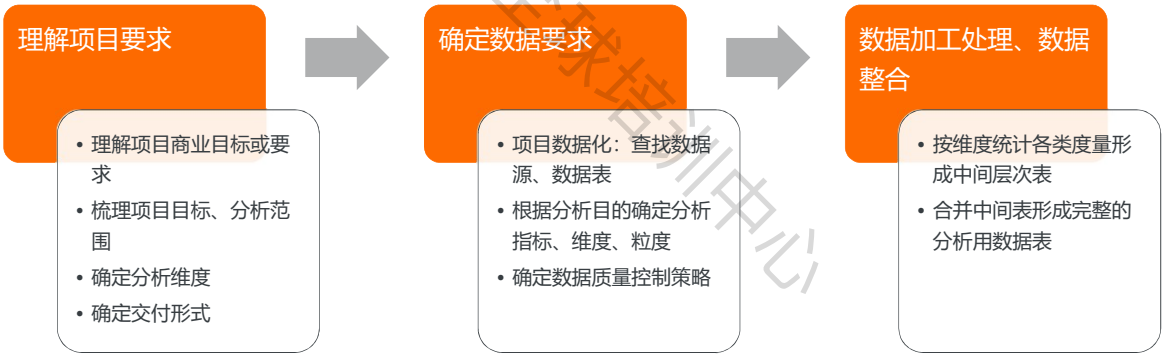
课程目录

1. **数据编程综述**
2. 编程方法
3. 编程效率

什么是数据编程

- 直观上讲就是通过编定计算机程序来处理数据，借以实现数据加工、数据分析、数据应用目的，即数据分析过程中的数据相关处理：
  - 编程语言，即编程工具，如SQL语言
  - 数据对象，包括数据分析过程中所涉及到的各类数据
  - 分析平台，即数据存储、计算平台，分析程序运行环境（软、硬），如常见的数据库
  - 编程方法、编程技巧、编程逻辑
  - 业务知识、数据中的业务含义、业务逻辑

数据分析编程的一般过程



数据分析编程的效率与质量问题

- 数据编程过程中一般涉及效率和质量问题：
  - 效率一般涉及编程效率和程序运行效率；
  - 编程效率：
    - ✓ 主要体现在编程能力
  - 编程规范性：
    - ✓ 统一的编程规范，规范的代码逻辑清晰、易于理解；
    - ✓ 程序易于复用；
    - ✓ 有利于程序的管理和维护等
  - 编程质量控制：
    - ✓ 包括多个方面，如运行的效率、数据的质量、程序的规范性、程序语法错误、逻辑错误等。
    - ✓ 核心是要保证分析结果的准确性，这是项目提出方的主要质量评估标准

课程目录

1. 数据编程综述
2. **编程方法**
3. 编程效率

数据分析的实现过程

- 数据分析的作用一般是现状分析、原因分析、预测分析，分析从需求开始到最终得出结论、提供报告为止，中间过程都涉及数据编程处理，数据处理过程如下：

数据分析编程方法

- 数据分析编程的实现方法-**倒推法**:

① 项目的商业目标或要求是什么?
② 应该建一个什么样的分析用数据表?
③ 应该如何建这样分析用数据表?

假设，业务部门需要你提交如下的报表:

超计划使用 # 月份	客户流失	客户未流失	总计	客户流失率
0	10,950	632,083	643,033	**1.7%**
1	2,410	123,649	126,059	**1.9%**
2	1,294	56,574	57,868	**2.2%**
3	589	26,189	26,778	**2.2%**
4	269	11,300	11,569	**2.3%**
5	143	6,637	6,780	**2.1%**
TOTAL	**15,655**	**856,432**	**872,087**	**1.8%**

数据分析编程方法

- 充分理解需求及分析目标常见方法 5w2h (又叫七问分析法):
  - **5w的内容:**
    ①why——为什么? 为什么要这么做? 理由何在? 原因是什么?
    ②what——是什么? 目的是什么? 作什么工作?
    ③where——何处? 在哪里做? 从哪里入手?
    ④when——何时? 什么时间完成? 什么时机最适宜?
    ⑤who——谁? 有谁来承担? 谁来完成? 谁负责?
  - **2h的内容:**
    ①how——怎么做? 如何提高效率? 如何实施? 方法怎么样?
    ②how much——多少? 做到什么程度? 数量如何? 质量水平如何? 费用产出如何?

超计划使用 # 月份	客户流失	客户未流失	总计	客户流失率
0	10,950	632,083	643,033	**1.7%**
1	2,410	123,649	126,059	**1.9%**
2	1,294	56,574	57,868	**2.2%**
3	589	26,189	26,778	**2.2%**
4	269	11,300	11,569	**2.3%**
5	143	6,637	6,780	**2.1%**
TOTAL	**15,655**	**856,432**	**872,087**	**1.8%**

数据分析编程方法

- 充分理解需求及分析目标：

梳理项目目标

主要问题：
- 客户流失率和超计划使用的#月份有什么关系？

问题分解：
- 统计每个顾客的超计划使用月数
- 统计每个顾客是否流失
- 按超计划使用月数分组，计算每一个组中的流失顾客数和非流失顾客数

梳理分析范围

分析范围通常指的是：
- 时间
- 地点
- 人或物

时间：
- 2016年1月至6月

地点：
- 全国范围

人或物：
- 活跃客户

认定分析维度

分析维度指的是：
- 最终分析报告中（需求结果中）以何种维度为重，维度关系到如何构建分析数据表

分析示例：
- 顾客行为分析，维度在顾客上，同时需要按月统计，分析数据表就以顾客ID和时间为复合主键

搞清项目交付形式

项目交付常见形式：
- 非正式汇报（Email）
- Word
- Excel
- PPT
- 标准IT项目

项目交付时间：
- Deadline

数据分析编程方法

- 理解需求及分析目标的基础上**确定数据需求:**

项目数据化

主要分析维度：
- 顾客ID

次要维度：
- 月，或时间相关

KPI涉及：
- 是否超过计划使用
- 是否流失

查找数据表

从主要维度开始查找：
- 主要维度通常都有自己的表，并拥有Key（主键、外键）
- 如果该表不存在，则使用事实表里的维度（例如交易的时间）

KPI计算使用事实表里的数据：
- 事实表要有主、次要维度信息
- 如果有多个事实表能用于计算KPI，则遵循以下原则：
  - ✓ 数据元组量最小
  - ✓ 数据精准度最高

数据质量控制

剖析、评估和清洗：
- 数据属性代表的意思能被用于计算相应KPI吗？
- 数据的缺失对分析结果有影响吗？
- 数据需要做转换吗？转换会损失精度吗？
- 分析项目通常需要多个表之间的整合才能获取到完整的信息，检查所需的Key是不是在相应表里都存在，并没有以下现象：
  - ✓ 缺失
  - ✓ 重复
  - ✓ 。。。

数据分析编程方法

- **分析用数据表（analytical table）**指将业务主题相关的指标、维度、属性关联在一起的一张数据库表。通常也叫**宽表，**从字面意义上讲就是字段比较多的数据库表。

  - **数据冗余：**由于把不同的内容都放在同一张表存储，宽表已经不符合三范式的模型设计规范，随之带来的主要坏处就是数据的大量冗余。

  - **性能优化：**这种宽表的设计广泛应用于数据分析和挖掘前的数据准备，通过把相关字段放在同一张表中，可以大大迭代计算时的效率问题。

数据分析编程方法

- **分析用数据表（analytical table）**指将业务主题相关的指标、维度、属性关联在一起的一张数据库表。通常也叫**宽表，**从字面意义上讲就是字段比较多的数据库表。

  - 根据业务部门的要求，为计算相应的KPI，一个如下所示的分析用数据表是合适的

年	月	客户ID	超过计划使用（0-未超过；1-超过）	流失与否（0-未流失；1-流失）
2016	1	2176981237213	1	0
2016	2	2176981237213	0	0
2016	3	2176981237213	0	0
2016	4	2176981237213	0	0
2016	5	2176981237213	0	0
2016	6	2176981237213	0	0
2016	1	6783264873246	0	0
2016	2	6783264873246	1	0
2016	3	6783264873246	1	1
2016	1	0923621123454	1	0
2016	2	0923621123454	1	0
2016	3	0923621123454	0	0

数据分析编程方法

年	月	客户ID	超过计划使用（0-未超过；1-超过）	流失与否（0-未流失；1-流失）
2016	1	2176981237213	1	0
2016	2	2176981237213	0	0
2016	3	2176981237213	0	0
2016	4	2176981237213	0	0
2016	5	2176981237213	0	0
2016	6	2176981237213	0	0
2016	1	6783264873246	0	0
2016	2	6783264873246	1	0
2016	3	6783264873246	1	1
2016	1	0923621123454	1	0
2016	2	0923621123454	1	0
2016	3	0923621123454	0	0

```
create temp table tmp_table as
select “客户ID”
, sum(“超过计划使用”) as “超计划使用#月份”
, max(“流失与否”) as “流失与否”
from analytical_table
group by “客户ID”;
```

客户ID	超计划使用#月份	流失与否
2176981237213	1	0
6783264873246	2	1
0923621123454	2	0

```
select “超计划使用#月份”
, sum(“流失与否”) as “客户流失”
, sum(case when “流失与否” = 0 then 1 else 0 end) as “客户未流失”
, count(distinct “客户ID”) as “总计”
, “客户流失” / “总计” as “流失率”
from tmp_table
group by “超计划使用#月份”;
```

超计划使用 # 月份	客户流失	客户未流失	总计	客户流失率
0	10,950	632,083	643,033	1.7%
1	2,410	123,649	126,059	1.9%
2	1,294	56,574	57,868	2.2%
3	589	26,189	26,778	2.2%
4	269	11,300	11,569	2.3%
5	143	6,637	6,780	2.1%
TOTAL	15,655	856,432	872,087	1.8%

数据分析编程方法

- 在这个例子里，分析用数据表有两种不同的列：
  - **维度**：“年”、“月”、“客户ID”都属于维度。
  - **度量**：“超过计划使用”、“流失与否”是度量。

年	月	客户ID	超过计划使用（0-未超过；1-超过）	流失与否（0-未流失；1-流失）
2016	1	2176981237213	1	0
2016	2	2176981237213	0	0
2016	3	2176981237213	0	0
2016	4	2176981237213	0	0
2016	5	2176981237213	0	0
2016	6	2176981237213	0	0
2016	1	6783264873246	0	0
2016	2	6783264873246	1	0
2016	3	6783264873246	1	1
2016	1	0923621123454	1	0
2016	2	0923621123454	1	0
2016	3	0923621123454	0	0

年	月	客户ID	应缴费	通话（分钟）	数据（M）	增值服务	消息	彩铃
2016	1	2176981237213	58.00	98	578	15	32	1
2016	2	2176981237213	62.48	104	983	15	41	1
2016	3	2176981237213	58.00	89	870	15	3	1
2016	4	2176981237213	58.00	78	750	15	13	1
2016	5	2176981237213	58.00	187	890	15	41	1
2016	6	2176981237213	98.56	56	1200	15	34	1
2016	1	6783264873246	128.00	678	300	15	1	0
2016	2	6783264873246	128.00	572	279	0	3	0
2016	3	6783264873246	128.00	498	240	0	0	0
2016	1	0923621123454	112.42	230	1357	0	6	0
2016	2	0923621123454	153.56	250	2367	0	8	0
2016	3	0923621123454	132.12	210	2980	15	9	0

从2016年1月份开始到6月，每个月的活跃客户都是分析项目的维度目标。

- ✓ 通常的情况下，数据仓库的维度表是第一考虑的数据源。
- ✓ 如果数据仓库里已经存在了关于客户活跃度的月度事实统计表，可以直接使用 。

```
create temp table analytical_table as
select distinct
“年”， “月”， “客户ID”
from “运营商月度服务汇总表”
where “通话” > 0 or “数据” > 0 or “消息” > 0;
```

数据分析编程方法

- 假设，事实表的数据质量符合要求，没有重复值和缺失值等现象，上述程序可以输出当月的活跃客户，**临时的分析用数据表**如下图所示

年	月	客户ID
2016	1	2176981237213
2016	2	2176981237213
2016	3	2176981237213
2016	4	2176981237213
2016	5	2176981237213
2016	6	2176981237213
2016	1	6783264873246
2016	2	6783264873246
2016	3	6783264873246
2016	1	0923621123454
2016	2	0923621123454
2016	3	0923621123454

接下来，分析师需要将**“超过计划使用“**的属性放到临时分析用数据表中。
为了判断客户在当月是否有超过计划使用的情况，有以下几种方式可以做到：

- 如果在数据仓库里已经有相关的数据表而且质量没有问题，可直接使用该表内关于客户在某年某月的“超过计划使用“属性（列）。
- 查找数据仓库里关于客户某年某月的套餐计划的实际可用量，与事实表整合后，比较”通话“、”数据“、”消息“等数据，实际用量超过计划允许的用量则客户在某年某月的“超过计划使用“属性（列）为1，否则为0。
- 也可以找到数据仓库里关于客户某年某月的计划合同价格，与事实表中的实际应付款相比较，实际应付款大于计划合同价格则“超过计划使用“属性（列）为1，否则为0。使用这种方法，必须要确保事实表中的实际应付款是代表了当月的使用量带来的费用，没有其他计费的考量，比如上月拖欠款、奖励款等都会导致数据标记出错。

数据分析编程方法

- 经过这一步数据处理，分析用数据表便加上了”超过计划使用“属性（列）。
  - 如下所示:

年	月	客户ID	超过计划使用（0-未超过；1-超过）
2016	1	2176981237213	1
2016	2	2176981237213	0
2016	3	2176981237213	0
2016	4	2176981237213	0
2016	5	2176981237213	0
2016	6	2176981237213	0
2016	1	6783264873246	0
2016	2	6783264873246	1
2016	3	6783264873246	1
2016	1	0923621123454	1
2016	2	0923621123454	1
2016	3	0923621123454	0

接下来，分析师需要将“**流失与否**”的属性加入分析用数据表中。通常运营商的数据仓库里会详细记录每个客户的开户日期和账户注销日期，通过账号注销的日期，分析师可以判断出客户在哪个月流失。源数据可以在客户产品关系表里找到，如下图所示：

套餐ID	套餐	客户ID	开通日期	关闭日期	数据（M）	通话（分钟）	消息	彩铃
000001	飞翔58	2176981237213	2014-10-23	2015-09-12	500	100	10000	1
000002	飞翔88	2176981237213	2015-09-12		1000	100	10000	1
000003	飞翔128	6783264873246	2015-10-23	2016-03-25	2000	200	10000	0
000001	飞翔58	0923621123454	2016-05-19		500	100	10000	0

数据分析编程方法

- 接下来，分析师需要将“流失与否”的属性加入分析用数据表中。源数据如下

套餐ID	套餐	客户ID	开通日期	关闭日期	数据（M）	通话（分钟）	消息	彩铃
000001	飞翔58	2176981237213	2014-10-23	2015-09-12	500	100	10000	1
000002	飞翔88	2176981237213	2015-09-12		1000	100	10000	1
000003	飞翔128	6783264873246	2015-10-23	2016-03-25	2000	200	10000	0
000001	飞翔58	0923621123454	2016-05-19		500	100	10000	0

```
create temp table tmp_table as select distinct
year(“关闭日期”) as “年”
, month(“关闭日期”) as “月”
, “客户ID”
, 1 as “流失与否”
from “客户产品关系表”
where “关闭日期”
between ‘2016-01-01’ and ‘2016-06-30’;
```

年	月	客户ID	客户流失（0-未流失；1-流失）
2016	3	6783264873246	1

阿里云全球培训中心

数据分析编程方法

- 最后，将tmp_table通过join聚合到分析用数据表，就完成了数据的整合，得到了完整的分析用数据表（analytical table）。需要注意的是，这样的聚合会产生大量的缺失值，需要在join结束后处理。

年	月	客户ID	超过计划使用（0-未超过；1-超过）
2016	1	2176981237213	1
2016	2	2176981237213	0
2016	3	2176981237213	0
2016	4	2176981237213	0
2016	5	2176981237213	0
2016	6	2176981237213	0
2016	1	6783264873246	0
2016	2	6783264873246	1
2016	3	6783264873246	1
2016	1	0923621123454	1
2016	2	0923621123454	1
2016	3	0923621123454	0

```
create temp table analytical_table
as select
a.*
, b.“流失与否”
from analytical_table as a left join tmp_table as b
on a.“年” = b.“年”
and a.“月” = b.“月”
and a.“客户ID” = b.“客户ID”;
```

年	月	客户ID	客户流失（0-未流失；1-流失）
2016	3	6783264873246	1

阿里云全球培训中心

数据分析编程方法

- 数据最终整合，基于整合数据生成最终结果。

年	月	客户ID
2016	1	2176981237213
2016	2	2176981237213
2016	3	2176981237213
2016	4	2176981237213
2016	5	2176981237213
2016	6	2176981237213
2016	1	6783264873246
2016	2	6783264873246
2016	3	6783264873246
2016	1	0923621123454
2016	2	0923621123454
2016	3	0923621123454

临时分析用数据表

增加度量

年	月	客户ID	超过计划使用（0-未超过；1-超过）
2016	1	2176981237213	1
2016	2	2176981237213	0
2016	3	2176981237213	0
2016	4	2176981237213	0
2016	5	2176981237213	0
2016	6	2176981237213	0
2016	1	6783264873246	0
2016	2	6783264873246	1
2016	3	6783264873246	1
2016	1	0923621123454	1
2016	2	0923621123454	1
2016	3	0923621123454	0

临时分析用数据表

增加度量

年	月	客户ID	超过计划使用（0-未超过；1-超过）	流失与否（0-未流失；1-流失）
2016	1	2176981237213	1	0
2016	2	2176981237213	0	0
2016	3	2176981237213	0	0
2016	4	2176981237213	0	0
2016	5	2176981237213	0	0
2016	6	2176981237213	0	0
2016	1	6783264873246	0	0
2016	2	6783264873246	1	0
2016	3	6783264873246	1	1
2016	1	0923621123454	1	0
2016	2	0923621123454	1	0
2016	3	0923621123454	0	0

分析用数据表

分组聚合

超计划使用 # 月份	客户流失	客户未流失	总计	客户流失率
0	10,950	632,083	643,033	1.7%
1	2,410	123,649	126,059	1.9%
2	1,294	56,574	57,868	2.2%
3	589	26,189	26,778	2.2%
4	269	11,300	11,569	2.3%
5	143	6,637	6,780	2.1%
TOTAL	15,655	856,432	872,087	1.8%

业务报表

课程目录

数据编程效率

- **编程效率**在工作中扮演着至关重要的角色，优秀的工程师往往都有着很高的效率。**常见方法**包括但不限于：
  - 训练编程能力：阅读一些诸如《Effective Java》和《Design Patterns》等类型的书，并每天做些基础训练
  - 阅读代码和技术资料：在当今开源世界的情况下，可以参阅大量优质代码。
  - 掌握代码编辑器：如Notepad++、VIM、Sublime Text、emacs、visual studio、IDEA等。
  - 善用搜索引擎：stackoverflow、csdn、google等。

数据取样

- 在数据编程中，通常编程速度的主要障碍是数据量太大，debug等待时间过长，解决方案是：
  - 用小样本数据进行编程
  - 可以由沙箱复制一个小样本数据仓库
  - 也可以由分析师根据分析项目的要求获取分析用小样本数据
- 在设计小样本数据仓库的时候，各个数据表不仅仅是从相对应的母表随机地取一部分样本那么简单。在取样的时候，数据表之间的数据逻辑关系还必须被完整保留，取数逻辑：
  1. 找到数据分析的主要维度，在维度表中首先进行随机取样；
  2. 以取样完成后的维度表为起始，找到事实表里与该维度相关的全部交易数据；
  3. 将第二步中事实表数据涉及到的维度数据抽样提取出来。

数据取样

- 企业级的数据库在取样时，由于要考虑多种分析的场景，通常采取更为复杂的方法来保证样本数据中客户行为、商品交易等业务数据的统计分布基本保持与母数据一致。

常见方法：

说明：当某个总体是由若干个有着自然界限和区分的子群（或类别、层次）所组成，同时，不同子群相互之间差很大、而每个子群内部的差异不大时，则适合于分层抽样的方法；反之，当不同子群之间差别不大、而每个子群内部的异质性比较大时，则特别适合于采用整群抽样的方法。

复用

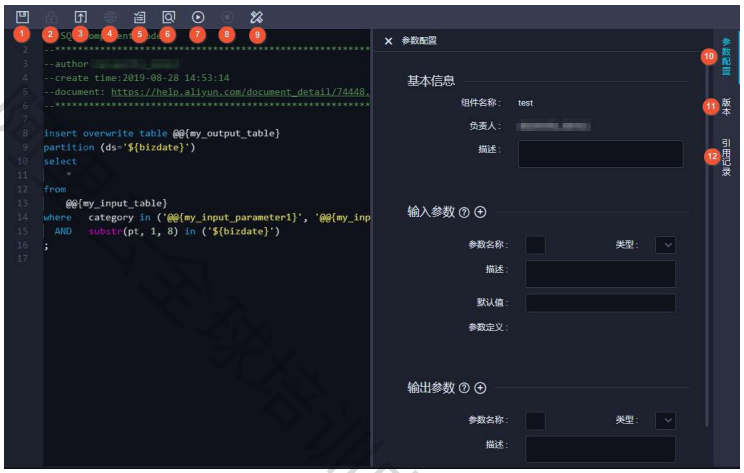
- **代码复用（code reuse）**：对曾经编写过的代码的一部分甚至全部重新加以利用，从而构建新的程序
  - **类型**
    - ✓ 机会性：开发团队在项目开始时便已知可复用的代码或组件的存在，因而决定进行代码复用。
    - ✓ 计划性：开发团队从战略角度为了日后的开发能够复用代码而进行设计。
  - **影响**
    - ✓ 可以减少代码量，也就潜在地减小了开发和维护成本。
    - ✓ 能力范围被约束住了，而且也增加了应用和被复用代码之间的耦合。

复用

- **继承（inheritance）**是面向对象的程序设计中（OOP）的核心基础的概念之一
  - 在程序设计中，当我们定义一个新类的时候，新的类称为子类（Subclass），而被继承的类称为基类、父类或超类（Base class、Super class）
  - 继承最大的好处是子类获得了父类的全部变量和方法的同时，又可以根据需要进行修改、拓展
  - 继承来源于现实世界，一个最简单的例子就是孩子会具有父母的一些特征，即每个孩子都会继承父亲或者母亲的某些特征，当然这只是最基本的继承关系，现实世界中还存在着更复杂的继承
- **影响**
  - 可以充分利用以前实现过的行为，减少开发时间。
  - 父类和子类都要进行足够的测试，否则会导致产生一个不易维护和扩展增强的脆弱继承结构。

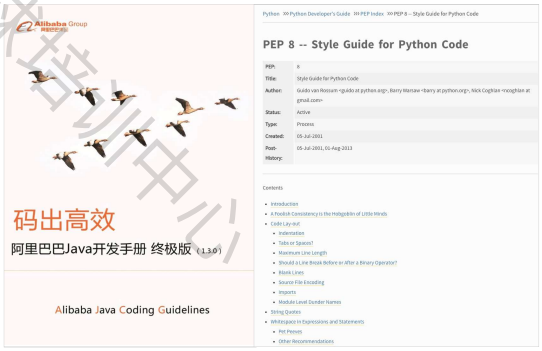
复用

- DataWorks提供的**组件**功能：一个组件就和一个函数的定义一样，由输入参数、输出参数和组件代码过程构成。



规范

- **编程规范**：即程序开发人员所编写源代码的书写风格、设计规范、设计原则 。
- 在企业内部实行统一的编程规范，能带来诸多好处：
  - 使代码更美观、阅读更方便。
  - 使代码的逻辑更清晰、更易于理解。
  - 增加程序被复用的机会。
  - 要素包括（但不限于）以下几点：
    - 缩进、注释、命名规范、代码设计开发原则规范等



规范

- **缩进（indent）**是管理代码块缩进以表达程序结构的一种约定
  - 缩进在大多数编程语言中（例如SQL）不是必要条件，而只是作为辅助符号。
  - 不过，部分语言（例如Python）使用缩进而非大括号或关键词来确定结构，这被称为越位规则。在这种语言中，缩进对编译器或解释器有意义，而不仅仅是清晰度或风格问题。

```
def is_even(a):
    if a % 2 == 0:
        print('Even!')
        return True
    else:
        print('Odd!')
        return False
```

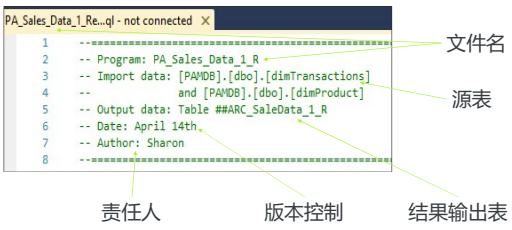
```
In [8]: is_even(4)
Even!
Out[8]: True

In [9]: is_even(3)
Odd!
Out[9]: False
```

说明：初学者（4个空格为语句块缩进）经常犯的错误是tab键和空间键混用，造成的缩进不一致。凡是看到这类报错：IndentationError，就是表示缩进错误或者不一致。

规范

- **注释（comment）**是对代码的解释和说明。作用是增强程序的可读性，可维护性，或者用于在源代码中处理不需运行的代码段，来调试程序的功能执行。
  - 块注释（ Block Comments ）通常适用于跟随它们的某些（或全部）代码，并缩进到与代码相同的级别。
  - 行内注释（Inline Comments）是与代码语句同行的注释，需要有节制的使用




```
/* if receive_flag is TRUE */
if (receive_flag);
```

注释意义不大

```
/* if mtp receive a message from links */
if (receive_flag);
```

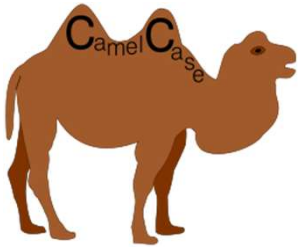
给出了额外有用的信息

规范

- **命名规范（Naming Conventions）**：当变量名和函数名称是由二个或多个单字链接在一起，而构成的唯一识别字时，可以利用“驼峰式大小写”来表示，可以增加变量和函数的可读性。
  - **小驼峰式**
    - ✓ 第一个单字以小写字母开始第二个单字的首字母大写。

      例如：firstName、lastName
  - **大驼峰式：**
    - ✓ 每一个单字的首字母都采用大写字母。

      例如：FirstName、LastName、CamelCase，也被称为Pascal命名法（Pascal Case）
  - **其他命名规范**
    - ✓ 匈牙利命名法



规范

- 数据分析SQL编程规范

字符集规范

库表结构规范

索引设计

字段类型设计规范

SQL代码编写规范

上线发布规范

规范

- **设计规范**：不同分析工具、不同运行平台有不同的编写、设计规范，规范设计开发的代码运行效率高、消耗资源少
  - 如MaxCompute SQL表和分区的设计原则、规范：
    - 所有的表和字段名要使用统一的命名规范。命名要求如下：
      - ✓能区分该表的业务类型。
      - ✓能区分该表是事实表、维度表、日志表或极限存储表。
      - ✓能区分该表的实体信息。
    - 不同表中具有相同业务含义的字段要定义成统一的数据类型，避免不必要的类型转换。
    - 分区设计及使用规则如下：
      - ✓支持新增分区，不支持新增分区字段。
      - ✓单表支持的分区数量上限为6万个。
      - ✓对于多级分区的表，如果想添加新的分区，则必须指明全部的分区值。
      - ✓不支持修改分区列的列名，只能修改分区列对应的值。
      - ✓修改多级分区的一个或者多个分区值时，多级分区的每一级的分区值都必须写上。

规范

- **开发规范**：不同分析工具、不同运行平台有不同的编写、设计规范，规范设计开发的代码运行效率高、消耗资源少
  - 示例，如MySQL的开发规范：
    - ✓ SELECT 语句必须指定字段名称，避免写*，减少数据读取量
    - ✓ INSERT 语句指定具体字段名称；
    - ✓ 尽量避免全表的扫描处理，加上必要的过滤条件；
    - ✓ WHERE条件里等号左右字段类型需一致，避免隐式转换，否则无法利用索引；
    - ✓ 使用LIKE 操作，左边不能加“%”，否则无法使用索引；
    - ✓ 索引列不能使用函数表达式
    - ✓ 不建议使用子查询
    - ✓ 避免多表关联
    - ✓ 选择合理的索引

阿里云

数据可视化

阿里云大数据分析师ACP认证课程

阿里云

课程目标

学习完本课程后，你将能够：

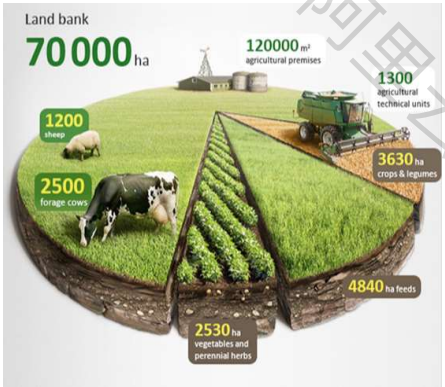
1. 了解可视化的基本知识，如定义、特点、实现方式等
2. 熟悉数据可视化常用图表，如特点和适用场景
3. 熟悉可视化产品的分类和基本设计原则
4. 掌握Quick BI产品特点、使用场景、使用方法
5. 掌握DataV产品特点、使用场景、使用方法

阿里云

课程目录

数据可视化的概念

数据可视化：将相对复杂的数据通过可视的、交互的方式进行展示，从而形象、直观地表达数据蕴含的信息和规律。



数据可视化作用

1、将事物图形化

2、将事物数值图形化

3、将事物关系图形化

4、将时间和空间可视化

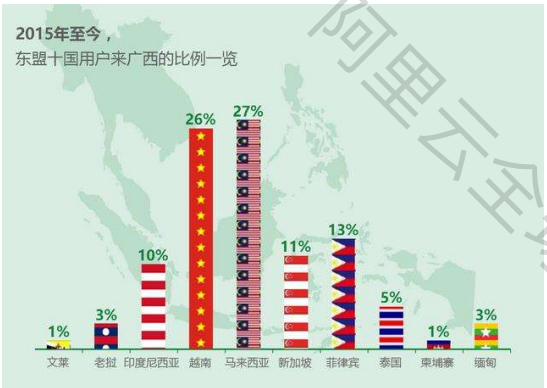
5、将概念进行转换

6、将事物抽象化

阿里云

将事物图形化

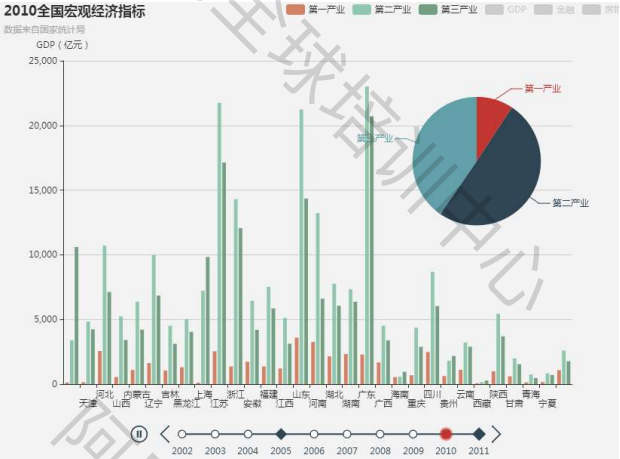
利用图形表示事物，方便看图人员，传递信息。



阿里云

将事物的数值图形化

每个事物的数值都是一个数据，将数据的大小以图形的方式表现。



将事物的关系图形化

当存在多个指标时，挖掘指标之间的关系，并将其图形化表达，可提升图表的可视化深度借助已有的场景表现通过构建场景表现。

图1 借助宇宙星系的场景来表现操作系统的分布

图2 构建颁奖台的场景来表现支付宝的年度账单

将时间和空间可视化

通过时间的维度来查看指标值的变化情况。用地图将空间可视化，地图作为主背景呈现所有信息点。

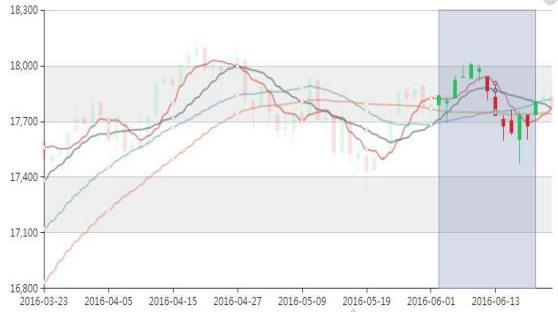


图1 将时间进行可视化

图2 将空间进行可视化

将概念进行转换

对数据的大小难以感知时，通常进行概念转换。

常用方法有对比和比喻

- 对比：通过对比来让用户感知数据
- 比喻：通过比喻手法来让用户感知数据

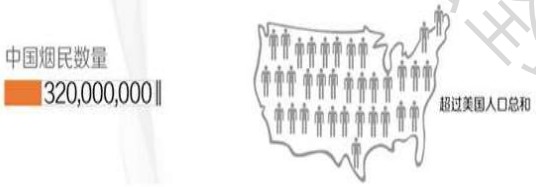


图1 中国烟民数量展示图

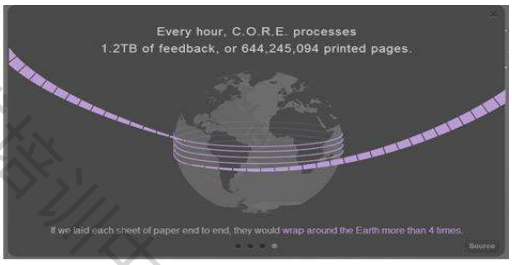
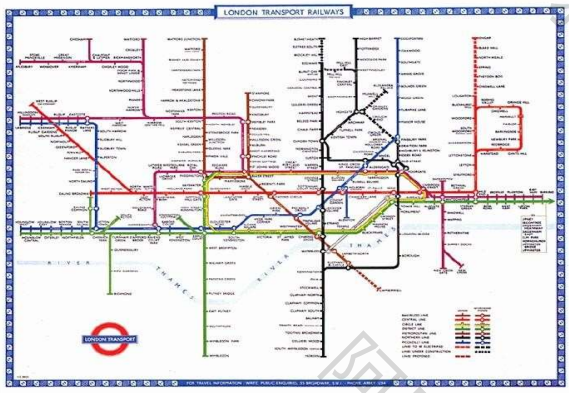


图2 雅虎邮箱处理的数据量

将事物抽象化

从具体事物抽出、概括出它们共同的方面、本质属性与关系等，而将个别的、非本质的方面、属性与关系舍弃



- 1933年Harry Beck设计的伦敦地铁图成为地铁路线的标准可视化方法。
- 颜色区分路线
- 水平、垂直、45度角
- 路线上的车站距离与实际距离不成比例关系

数据可视化的价值

数据可视化的核心价值：

- 视觉吸引力强
- 易于理解
- 容易记住
- 总结信息之间的关系。

课程目录

阿里云

数据可视化常见图表类型


线图


柱图


饼图


散点图


雷达图


漏斗图


旋风漏斗


树图


矩阵树图


来源去向

指标看板


仪表盘


地图


极坐标图

词云

阿里云

折线图

- **折线图**：又称为线图，指通过直线将一些数据点按照某种顺序连起来形成的图。

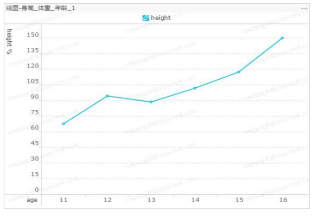
- **场景**：查看数据中随一个有序变量（如时间）变化的趋势。

  例如：五年期的股价变化、一个月内的网页查看数、逐季收入增长情况。

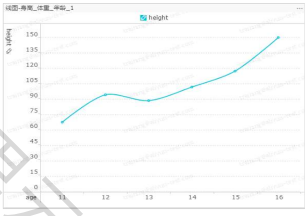
  - 可以清晰展现数据的增减的趋势、速率、规律、数据的峰值和谷底等特征。
  - 每张图表上折线条数不宜过多。

- **类似图表**：堆积图、曲线图、双Y轴折线图、面积图

折线图 - 展示形式

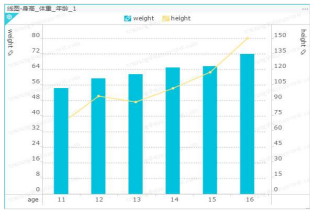


线图

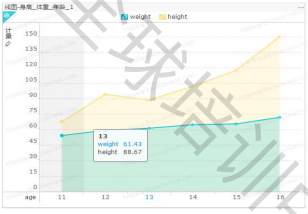


曲线图

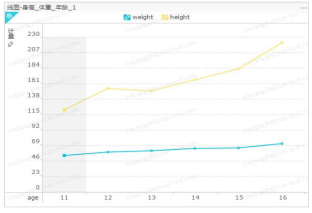
多指标折线图



双Y轴图



面积图



堆积图

柱状图

- **柱状图**：又称为柱图，指按长方形的长度来表达数值的统计报告图，使用垂直或水平的柱子显示类别之间的数值比较。其中一个轴表示需要对比的分类维度，另外一个轴代表相应的数值。

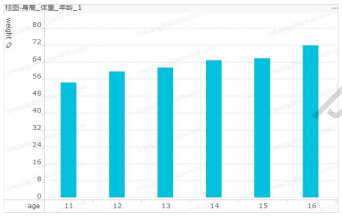
- **场景**：适合用于跨类别比较数据。

  例如：不同类型客户的数量、按来源站点划分的网站流量、按分区划分的消费比率。

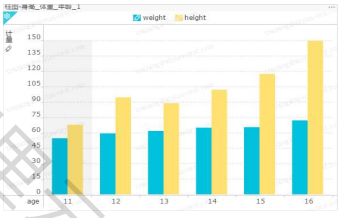
  - 简单直观，可迅速做出比较，一目了然地揭示高低点
  - 不适合较大的数据集展现

- **类似图表**：条形图、直方图、堆积图、百分比堆积图、双Y轴等。

柱状图 - 展示形式



柱图

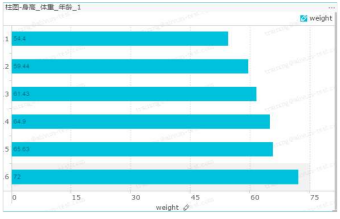


分组柱图

堆积图

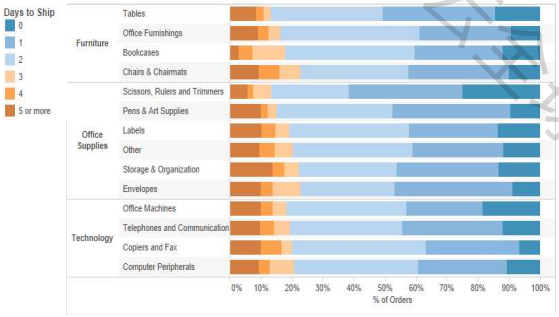
百分比堆积图

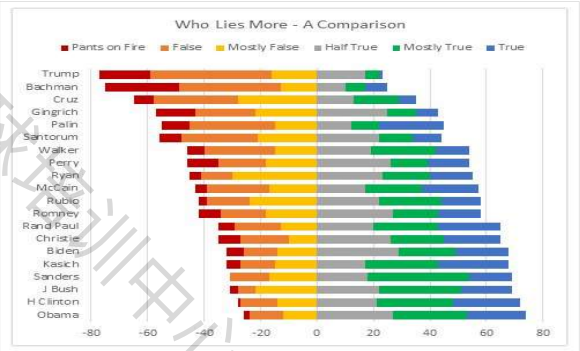
双Y轴柱图



条形图

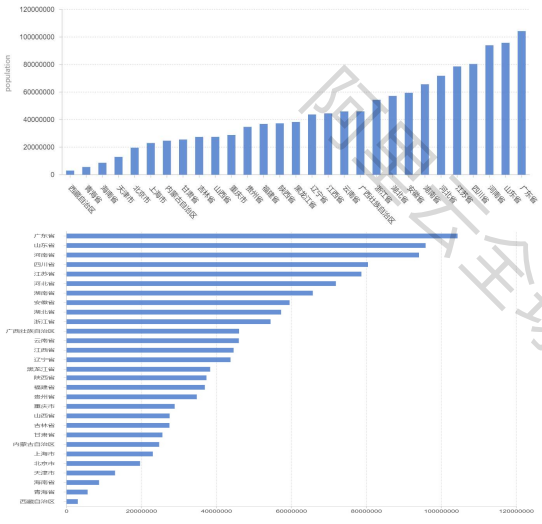
柱状图 - 展示形式特殊说明





- **上下或左右并列显示**：能够深化分析，一次解决多个问题。
- **给条形加上色彩**：获得更好的效果，用层叠的色彩能立刻带来洞见。
- **把条形放在数轴的两侧**：把正负数据点沿着连续轴画出来，是发现趋势的有效方式。

柱状图 - 展示形式特殊说明



- 分类情况过多时，分类的文本为了排布合理，需要进行旋转，不利于阅读。
- 纵向柱状图最好不超过12条数据。

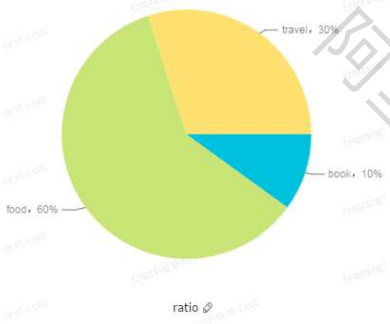
- 条形图最好不超过 30 条数据。

饼图

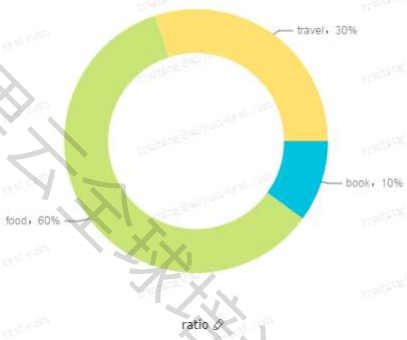
- **饼图**：又称扇形图，指以饼状图形显示一个数据系列中各项的大小与总和的比例。

- **场景**：用于表示不同分类的占比情况，通过弧度大小来对比各种分类。
  - 简单直观，很容易看到组成成分的占比。
  - 饼图不适用于多分类的数据。
  - 相比于具备同样功能的其他图表（如百分比柱状图），饼图需要占据更大的画布空间。
  - 很难进行多个饼图之间的数值比较。
- **类似图表**：环形图。

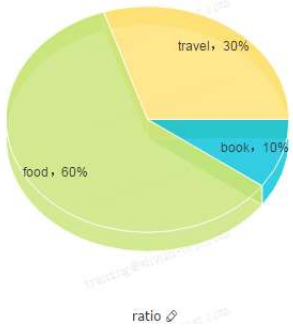
饼图 - 展示形式



饼图



环图



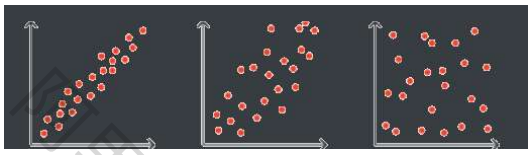
3D饼图

散点图

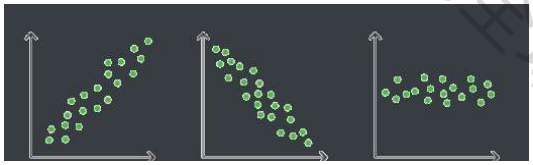
- **散点图**：也叫 X-Y 图，它将所有的数据以点的形式展现在直角坐标系上，以显示变量之间的相互影响程度，点的位置由变量的数值决定。

- **场景**：显示和比较数值。
  - 显示趋势，还能显示数据集群的形状，以及在数据云团中各数据点的关系。
  - 适合展示较大的数据集。
  - 看上去比较杂乱，除了数据的相关性和分布情况，其他信息不能很好展现。
- **类似图表**：气泡图。

散点图 - 展示形式



强　　弱　　无



正相关　　负相关　　不相关

线性　　指数　　U形

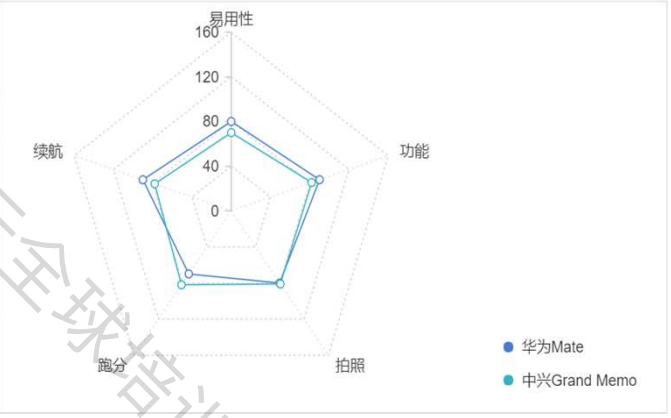
雷达图

- **雷达图**：又叫蛛网图，将多个维度（一般>4个）的数据量映射到坐标轴上，这些坐标轴起始于同一个圆心点，通常结束于圆周边缘，将同一组的点使用线连接起来。

- **场景**：适用于多维数据集。
  - 适合展现某个数据集的多个关键特征，以及和标准值的比对。
  - 适合比较多条数据在多个维度上的取值。
  - 多边形过多会使可读性下降，使整体的图形过于混乱。特别是当对多边形进行颜色填充后，上层会遮挡覆盖住一部分甚至全部的下层多边形。
  - 如果变量过多，也会造成可读性下降，因为一个变量对应一个坐标轴，这样会使坐标轴过于密集，使图表给人感觉很复杂。
- **类似图表**：无。

雷达图 - 展示形式

适合的数据	N个分类字段，一个连续字段(N <= 8)
功能	对比分类数据的数值大小
数据与图形的映射	一个分类字段映射到极坐标下角度标定的位置。另一个分类字段映射到颜色，连续字段映射到半径长度。
适合的数据条数	不超过30*N条（N是雷达链分类数）



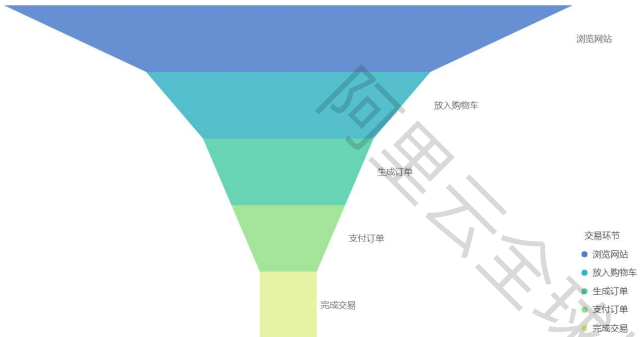
- 雷达图每个轴线都表示不同维度，但使用上为了容易理解和统一比较，经常会人为的将多个坐标轴都统一成一个度量，比如：统一成分数、百分比等。

漏斗图

- **漏斗图**：有多个梯形从上而下叠加而成。适用于业务流程比较规范、周期时间长、环节多的单流程单向分析，通过漏斗中各环节业务数据的比较，能够直观地发现和说明问题所在的环节，进而做出决策。

- **场景**：常用的是转化率和流失率两个互补型指标。
  - 漏斗图总是开始于一个100%的数量，结束于一个较小的数量。
  - 在开始和结束之间由N个流程环节组成。
  - 每个环节用一个梯形来表示，梯形的上底宽度表示当前环节的输入情况，梯形的下底宽度表示当前环节的输出情况，上底与下底之间的差值形象的表现了在当前环节业务量的减小量，当前梯形边的斜率表现了当前环节的减小率。
- **类似图表**：金字塔图、对称漏斗图（旋风）、对比漏斗图

漏斗图 - 展示形式



适合的数据	一个分类数据字段、一个连续数据字段
功能	对比分类数据的数值大小
数据与图形的映射	分类数据字段映射到颜色 连续数据字段映射到梯形的面积
适合的数据条数	不超过12条数据

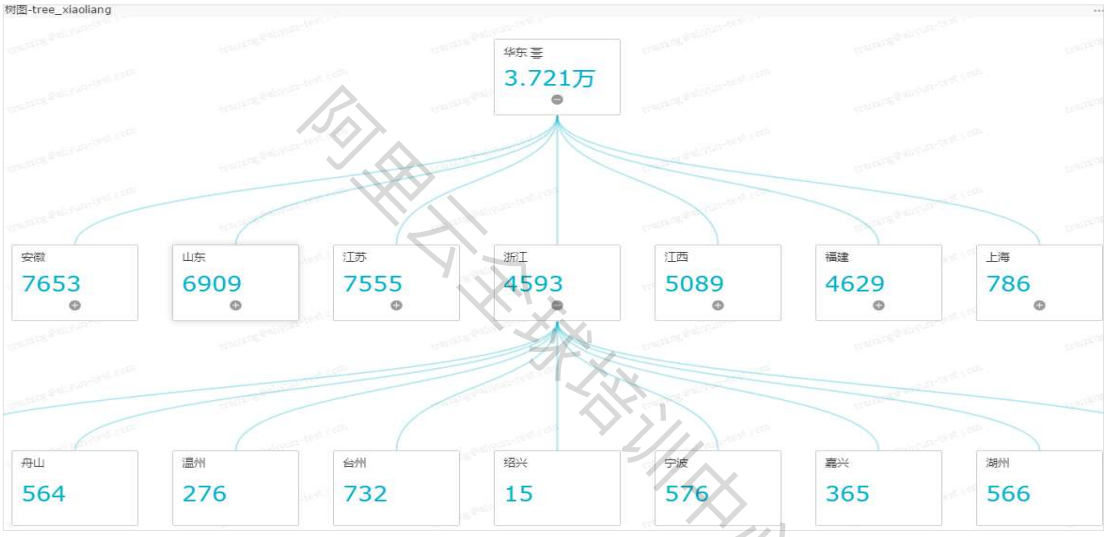
- 通过给不同的环节标以不同的颜色，可以帮助用户更好的区分各个环节之间的差异。注意，所有环节的流量都应该使用同一个度量。

树图

- **树图**：树图是通过树形结构来展现层级数据的组织关系，以父子层次结构来组织对象，是枚举法的一种表达方式。

- **场景**：适用于与组织结构有关的分析，即有明确的层次关系的数据。
  - 直观的展现层次关系。
  - 可以看到各层级指标间的关系，可进行简单的上卷、下钻等操作。
  - 数据层级不宜过多，每层的成员不宜过多。
  - 无法展现各部分占比关系。
- **类似图表**：矩阵树图。

树图 - 展示形式



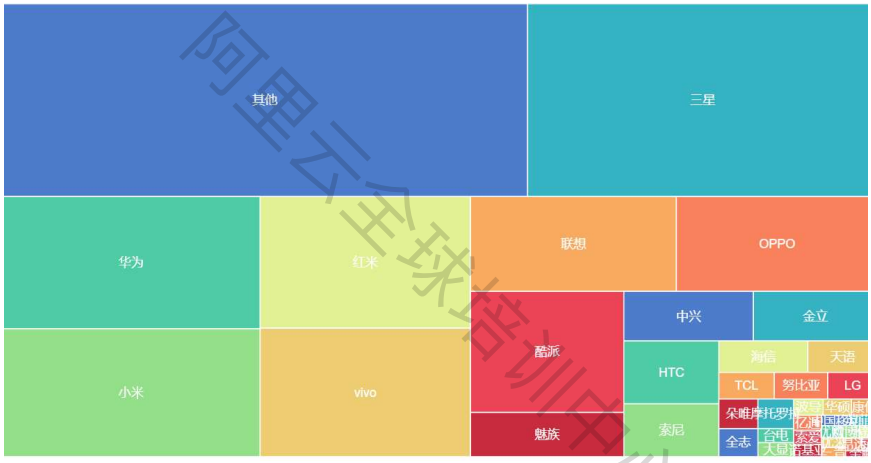
矩阵树图

- **矩阵树图：**采用矩形表示层次结构的节点，父子层次关系用矩形间的相互嵌套来表达。从根节点开始，空间根据相应的子节点数目被分为多个矩形，矩形的面积大小对应节点的属性。每个矩形又按照相应节点的子节点递归的进行分割，直到叶子节点为止。

- **场景**：适合展现具有层级关系的数据，能够直观体现同级之间的比较。
  - 图形更紧凑，同样大小的画布可以展现更多的信息。
  - 可以展现成员间的权重。
  - 不够直观、明确，不像树图那么清晰。
  - 分类占比太小时不容易排布。
- **类似图表**：树图、马赛克图、热力图。

矩阵树图 - 展示形式

某年手机品牌销量矩阵树图

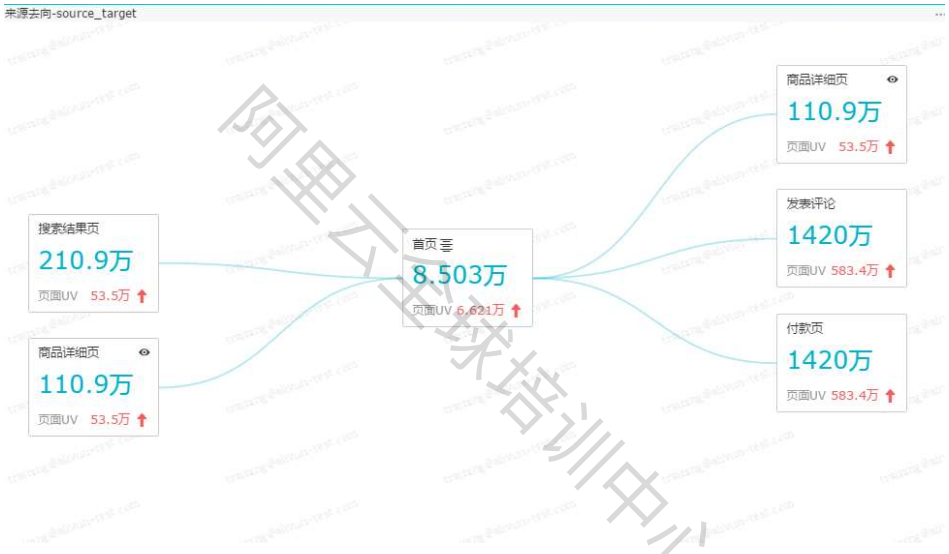


来源去向图

- **来源去向图**：通过页面访问量 PV 和访客的数量 UV推算出网页的转化率，进而可以了解网站的整体运营效果和某一类商品的最终成交量。

- **场景**：适用于电商或与营销有关系的分析，比如分析购物网站中，哪些商品最畅销或者哪一个时间段是访问高峰。
  - 特别适合分析展现网站流量的运营数据。
  - 显示结果直观，可以清晰的看到各个维度指标变化的情况。
  - 支持以某个节点查看该节点所在流程的情况。
  - 应用面很窄，只能显示三级维度的流程数据。
  - 对显示的度量要求严格。
- **类似图表**：无。

来源去向图 - 展示形式



指标看板

- **指标看板**：通过文字、数字和符号的合理排版，对数据进行一目了然的展示。由看板标签和看板指标组成，标签由维度决定，指标由数据的度量决定。

- **场景**：适合用来展示一个维度下的一个或者多个度量，特别是对某些指标需要精确读数的场景。
  - 展现的是详细的数字，用户得到的都是精确信息。
  - 简单直观，重点数字突出，容易得到关键信息。
  - 展现维度只能有一个。
  - 展现指标不宜过多。
  - 只是数字面板，不具有图形的各种优势。
- **类似图表**：无。

指标看板 - 展示形式

仪表盘

- **仪表盘**：像一个钟表或者刻度盘，有刻度和指针，其中刻度表示度量，指针表示维度，指针角度表示数值，指针指向当前数值。

- **场景**：管理报表或报告，直观的表现出某个指标的进度或实际情况。
  - 将专业数据通过常见的刻度表形式展现，非常直观易懂。
  - 拟物化的展现更人性化。
  - 适用场景比较窄，主要用于进度或占比的展现。
  - 只能一个维度，指标也不宜过多，展示信息有限。
- **类似图表**：堆积图

仪表盘 - 展示形式

地图

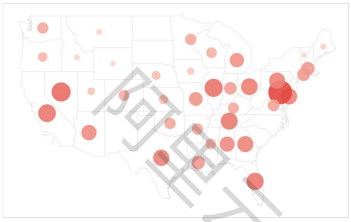
- **地图**：指通过地图作为背景，通过图形的位置来表现数据的地理位置，同时将数据通过颜色或者气泡映射在不同的地理位置上。

- **场景**：适合带有地理编码的数据展现。例如：按州划分的保险索赔、按国家划分的出口目的地、按邮政编码划分的车祸、自定义销售区域。
  - 对数据的地理分布显示直观。
  - 通过颜色深浅、气泡大小等容易判断度量的大小。
  - 必须有地理信息，且数据为汇总数据，气泡容易叠加。
  - 气泡大小和颜色深浅相近时，不易分辨。
  - 地理面积大小和度量值无关，容易误读。
- **类似图表**：气泡地图、颜色地图（分级统计地图）、点描法地图。

地图-展示形式

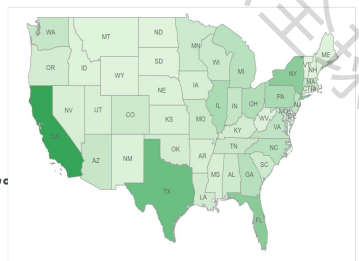
气泡地图

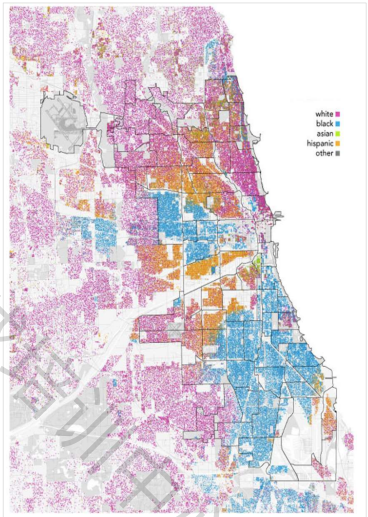
- 美国各州在某年发生的抢劫案件数目。



颜色地图

- 2014 年美国各个州的人口情况。





点描法地图

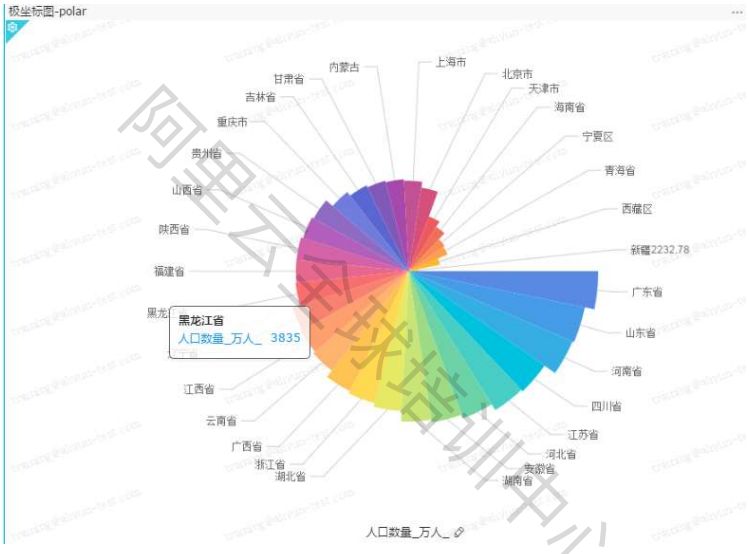
- 2010 年芝加哥人口种族分布。

极坐标图

- **极坐标图**：由多个扇区构成的。每个扇区的标签由数据的维度决定，每个扇区长度由数据的度量决定。每个扇形的角度一样，通过半径展示变化。

- **场景**：适用于枚举数据之间的比较，比如显示一段时间内的数据变化，或显示各项之间的比较情况。
  - ■ 视觉效果在部分情况下比其他图表更佳。
  - ■ 同样的画布能比部分其他图表展示数据更多。
  - □ 不适合分类过少的数据集。
  - □ 不适合部分度量值过小的数据集。
- **类似图表**：饼图、环图、柱图、玫瑰图等。

极坐标图-展示形式



词云图

- **词云图**：又称文字云，是文本数据的视觉表示，由词汇组成类似云的彩色图形，用于展示大量文本数据。每个词的重要性以字体大小或颜色显示。
- **场景**：适合用于描述网站上的关键字（即标签），或可视化自由格式文本，可以对比文字的重要程度。其本质是点图，是在相应坐标点绘制具有特定样式的文字的结果。
  - 快速感知最突出的文字，或区别权重不同的文字。
  - 可展示大量文本。
  - 不适合展现的数据太少的数据集。
  - 不适展现区分度不大的数据，即无重点关键词。
- **类似图表**：点图、柱图。

词云图-展示形式

图表类型核心要素

图表名称	图表数据要素	数据要素构成
线图	类别轴，值轴	类别轴上最少1个纬度；值轴上最少1个纬度。
柱图	类别轴，值轴	类别轴上最少1个纬度；值轴上最少1个纬度。
饼图	扇区标签，扇区角度	扇区标签上有且仅有1个纬度，并且纬度值小于等于12；扇区角度上有且仅有1个度量。
气泡地图	地理区域，气泡大小	地理区域有且仅有1个纬度，并且为地理纬度；气泡大小由最小1个最多5个度量。
色彩地图	地理区域，色彩饱和度	地理区域有且仅有1个纬度，并且为地理纬度；色彩饱和度最少1个最多5个度量。
仪表盘	指针角度，工具提示	有且仅有1个度量。
雷达图	分支标签，分支长度	分支标签最少1个最多2个纬度，分支长度最少1个度量。
散点图	颜色图例，X轴，Y轴	颜色图例有且仅有1个纬度，并且纬度成员的数量可达1000；X轴：最少1个最多3个度量；Y轴：有且仅有1个度量。

图表类型核心要素

图表名称	图表数据要素	数据要素构成
漏斗图	漏斗层标签，漏斗层宽	漏斗层标签有且仅有1个纬度；漏斗层宽有且仅有1个度量。
指标看板	看板标签，看板指标	看板标签最多1个纬度；看板指标最少1个，最多10个度量。
矩阵树图	色块标签，色块大小	扇区标签上有且仅有1个纬度，并且纬度值小于等于12；扇区角度上有且仅有1个度量。
极坐标图	扇区标签，扇区长度	扇区标签上有且仅有1个纬度，并且纬度值大于等于3且小于等于12；扇区长度上有且仅有1个度量。
词云图	词大小，词标签	词大小有且仅有1个纬度；词标签有且仅有1个度量。
旋风漏斗	对比主题，对比指标	对比主题有且仅有1个纬度；对比指标最少1个度量。
树图	树父子节点标签，树父子节点指标	树父子节点标签最少2个纬度；树父子节点指标最少1个度量。
来源去向	前一页面，当前页面，后一页面；前一页面PV，前一页面UV，当前页面PV，当前页面UV，下一页面PV，下一页面UV，路径转化率，页面跳出率	所有数据要素，分别只取1个纬度或者1个度量。

图表类型和使用场景

比较	对比各个值之间的差别	柱图	雷达	漏斗	极坐标	词云
占比	部分占整体的百分比	饼图	漏斗	仪表盘	矩阵树图	
相关	显示各个值之间的关系	散点	指标看板	树图	矩阵树图	来源去向图
趋势	数值随维度的变化情况	线图	柱图			
地理图	数值和地理信息映射图	气泡地图	色彩地图	点描法地图		

课程目录

数据可视化设计步骤

1. **明确问题**

   图表要解决什么问题希望传达给用户什么内容。

2. **基本框架**

   对整个图表设计一个初步的框架，即概览图。

3. **确定指标**

   将最终需要提供的信息转换成明确的指标。

4. **确定图表类型**

   根据要展现的内容和信息选择合适的图标类型。

5. **突出关键信息**

   通过图表设计将用户的注意力引向关键信息。

可视化设计原则

可视化设计原则 - 突出变化

突出变化：快速掌握业务的变化，要将这个变化放在最突出醒目的位置，同时要考虑KPI的时效性。

- 零售业绩通常以周为单位
- 网站关心每小时的变化
- 政府关心每天的变化
- 双十一关心每秒的变化

可视化设计原则 - 引发提问

引发提问：KPI数值的变化能够直接导致业务部门提出问题。

- What（发生了什么）
- Why（为什么发生）
- So what（需要做什么）

可视化设计原则 - 保持一致

保持一致：帮助观众在理解KPI背后的故事的时候不易发生歧义

- 例如：主要KPI的单位为天，则其他辅助KPI的时间单位也为天。
- 左图中，底部有表平均容量和表存储容量，这两个指标的单位应该保持一致，否则很容易造成数据的误读。

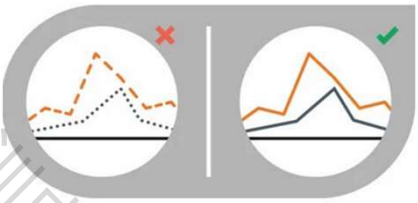
可视化设计原则 - 美观易懂

美观易懂：包括颜色的使用，各个模块和图表的位置、大小和他们之间的逻辑关系等等。

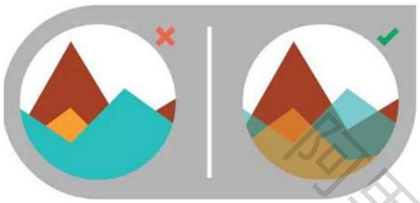
可视化设计常见错误

混乱的饼图分割

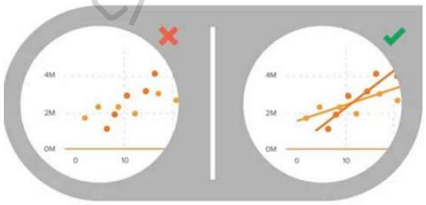
在折线图中使用不连贯的线条



数据模糊不清



让读者自己解读



可视化产品常见分类

数据可视化常见产品主要分为两大类，一类用于BI报表制作与数据分析，一类用于制作数据可视化大屏。

1. BI报表制作类

- 是向企业和个人提供的高效数据分析及展现服务平台，承载了数据连接、数据处理、数据分析及可视化的能力。
- 相关产品：阿里云Quick BI

2. 可视化大屏类

- 是将数据显示在超大屏幕上，包括集团会议大屏、管理驾驶舱、高层办公室会议室。它对实时性要求很高，会更加注重性能的数据显示，这是为什么数据可视化大屏会流行未来科幻风格，追求震撼的视觉效果。设计3D的动态效果，也经常用一些交互设计，以显示数据之间的联系。
- 相关产品：阿里云DataV

课程目录

Quick BI是什么

Quick BI是一个简单易用的BI报表制作与分析工具。基于云计算，致力于大数据高效分析与展现的轻量级**自助BI工具服务平台**。通过对数据源的连接和数据集的创建，对数据进行**即时的分析与查询**；通过电子表格或仪表板功能，以拖拽的方式进行数据的**可视化呈现**。

多种数据源	易用	灵活	安全
• 支持云数据库、关系型数据库、Hadoop、MPP等数据源接入	• 拖拉拽操作 • 丰富可视化图表 • 强大建模	• 可嵌入自有系统 • 实现免登陆	• 组织管理 • 行级权限 • 工作空间隔离

Quick BI产品定位及业务价值

- Quick BI 提供海量数据实时在线分析，拖拽式操作、丰富的可视化效果，让懂业务的人员自助实现数据分析、重塑数据生产的全链路，最终实现人人都是数据分析师。

节省人力成本

降低企业专业人才依赖

业务响应提效

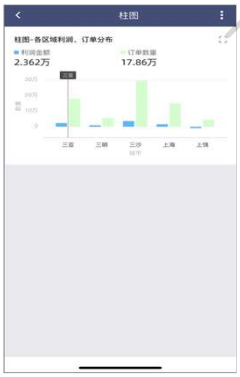
效率及业务分析思维的提升

人人都是分析师

企业全链路数据决策实现

Quick BI 产品特性：支持多端适配展现

- 一次配置，自动适配移动端、PC端、移动端采用分布式自适应分布展现。



Quick BI 产品特性：支持钉钉微应用集成

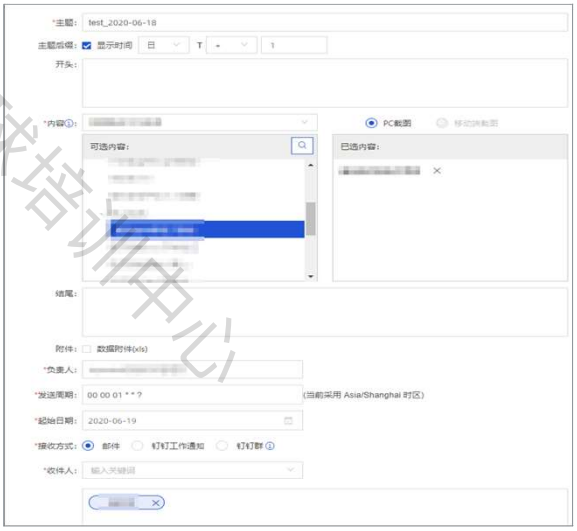
- 支持和钉钉账号绑定，通过钉钉微应用在移动端便捷访问数据门户及相关数据表、进行权限审批等操作。

Quick BI 产品特性：支持在线协同及分享机制

- 实现在线协同编辑、并通过公开、分享等方式，让数据分析服务快速、便捷地在线化运转。

Quick BI 产品特性：支持邮件订阅

- 通过内容截图方式，自动推送仪表板或电子表格，满足日常周报、月报等推送的诉求。



阿里云

课程目录

阿里云

数据即时分析与决策

解决的问题：

- 取数难

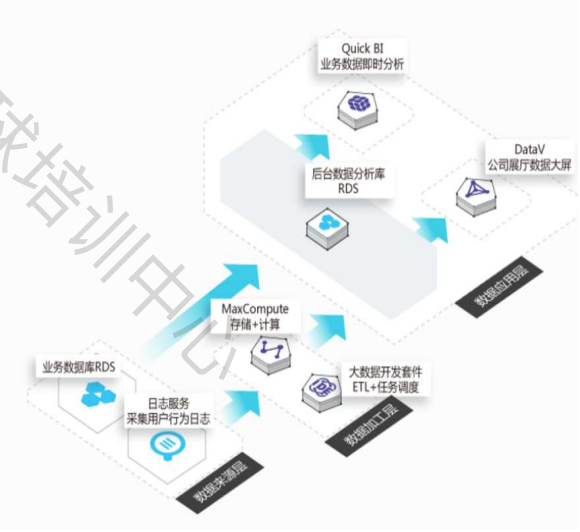
  业务人员需经常找技术写SQL取数查看各个维度的数据做决策。

- 报表产出效率低，维护难

  后台分析系统的数据报表变更，编码研发周期长，维护困难。

- 图表效果设计不佳，人力成本高

  使用开源类图表工具做报表，界面效果不佳，人力维护成本高。



阿里云

报表与自有系统集成

优势及能力：

- 上手快

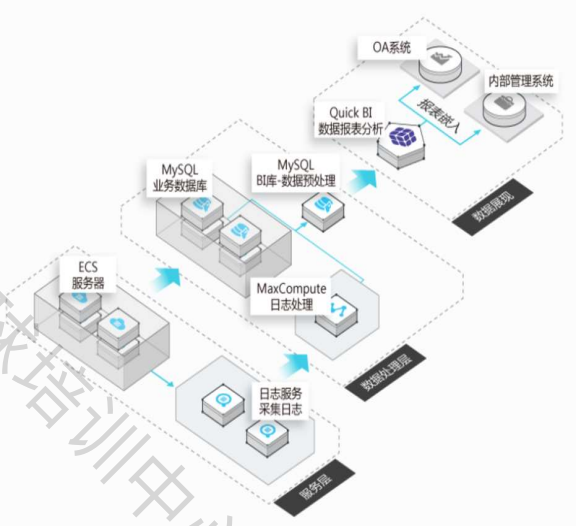
  上手简单，快捷，满足不同岗位的数据需求，学习门槛低。

- 极大提高看数据的效率

  与内部系统集成，可结合进行数据分析，极大提高看数据的效率。

- 统一系统入口

  解决员工使用多系统的麻烦，利于使用与控制。



阿里云

交易数据权限管控

功能说明：

- 数据权限行级管控

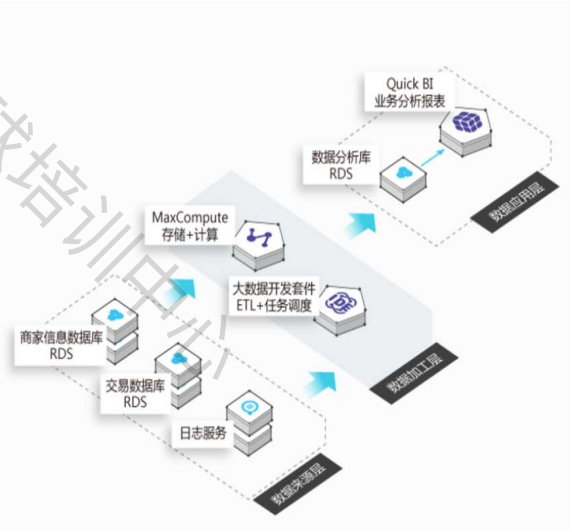
  轻松实现同一份报表，不同区经理只看到本区的相关数据。

- 适应多变的业务需求

  统计指标经常根据业务发展而频繁变动，负担重，响应慢。

- 跨源数据集成及计算性能保障

  充分利用云上BI的底层能力，解决跨源数据分析及计算性能瓶颈问题。



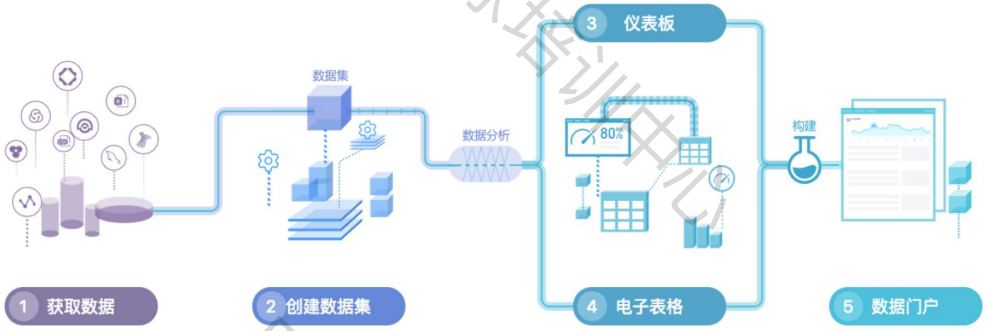
阿里云

课程目录

阿里云

Quick BI 核心主流程

- 工作空间下，包括数据源、数据集、仪表板、电子表格和数据门户5大功能，空间管理员和开发者可以进行数据源和数据集管理，分析师可以制作仪表板、电子表格和数据门户，阅读者即可阅览查看。



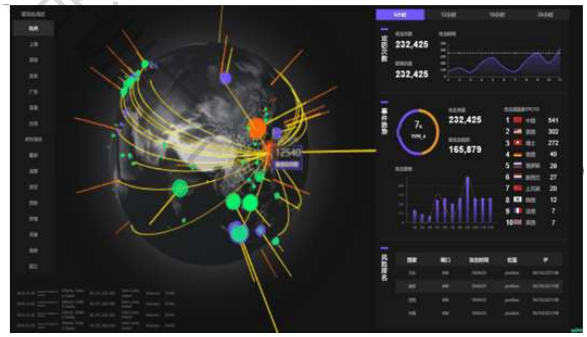
阿里云

课程目录

阿里云

DataV产品简介

DataV是使用可视化应用的方式来分析并展示庞杂数据的产品，旨在帮助非专业的工程师通过图形化的界面轻松搭建专业水准的可视化应用，满足会议展览、业务监控、风险预警、地理信息分析等多种业务的展示需求。



DataV产品特性

便捷搭建数据应用，赋能业务创新

- 模板丰富
- 配置简单
- 人性化图形界面
- 提供数据交互功能，图表联动

优秀的数据展示效果

- 多屏适配，一次编辑多端使用
- 组件库丰富，且支持第三方图标库
- 地理信息系统组件丰富，支持空间数据可视化分析

统一运维，安全可控

- 提供统一运维，减少人力输出
- 支持密码与token加密，集成访问权限体系
- 平台提供统一安全管控

成熟稳定，技术开放

- 支持二次开发，客户可以定制组件
- 支持多端发布，可跨多平台、多系统，支持各类业务系统

DataV产品特性-丰富的数据源接入支持

- 能够接入阿里云的分析型数据库和关系型数据库，支持本地CSV上传、在线API接入及动态请求。满足各类大数据实时计算、监控的需求，充分发挥大数据计算的能力。

数据库类

- Analytic DB
- RDS for MySQL
- RDS for PostgreSQL
- RDS for SQLServer
- Analytic DB for PostgreSQL
- TableStore
- Oracle
- 兼容MySQL数据库
- 对象存储OSS
- 交互式分析Hologres

文件类

- CSV
- 静态JSON

API类

- Open API
- 阿里云API网关

其它类

- DataV数据代理服务
- 业务实时监控服务ARMS
- 日志服务SLS
- DataWorks数据服务
- 阿里云Elastic search
- 区块链服务

DataV产品特性 - 多样化的场景模板支持

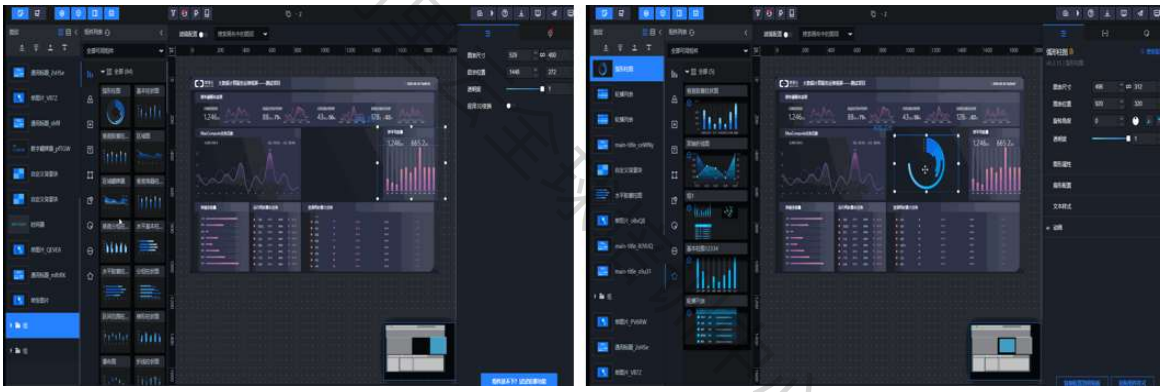
- 数据可视化的设计难点并不在于图表类型的多样化，而在于如何能在简单的一页之内让用户读懂数据之间的层次与关联，这就关系到色彩、布局、图表的综合运用。DataV提供指挥中心、地理分析、实时监控、汇报展示等多种场景模版，方便设计师进行可视化产品的设计和开发。

DataV产品特性 - 丰富的图表组件资源

- DataV除了针对业务展示优化过的常规图表外，还能够绘制包括海量数据的地理轨迹、地理飞线、热力分布、地域区块、3D地图、3D地球，实现地理数据的多层叠加。此外还有拓扑关系、树图等图表供自由搭配。

DataV产品特性 - 图形化的便捷搭建工具

- DataV提供多种业务模块级别而非图表组件的工具，所见即所得的配置方式，无需编程能力，只需要通过拖拽，即可创造出专业的可视化应用。



DataV产品特性 - 多分辨率适配与灵活的发布方式

- 针对拼接的可视化应用端的展示做了分辨率优化，能够适配非常规的拼接分辨率。创建的可视化应用能够发布分享，授权互联网用户也可以访问到应用，作为企业对外数据业务展示的窗口。

DataV产品特性 - 其它特性

- 支持本地化运行部署
- 具备大屏拼接中控系统
- 支持二次开发

课程目录

运营业务数据看板

- **数据看板**：通过DataV制作数据看板能够直观反映出业务变化，并有助于决策层发出业务调整与决策。

管理驾驶舱

- **管理驾驶舱**：为企业内部领导及相关高管提供指标分析型系统，让日常或者周期性的总结性指标实时或周期的展示在可视化屏幕上。

阿里云

应急指挥大屏

- **应急指挥大屏**：作为指挥中心大屏，适用于公安、交通等行业，提供应急指挥调度等。

阿里云

城市展厅

- **城市展厅**：配合模型沙盘、宣传影片、全程互动讲解，展示城市规划成果。

课程目录

DataV制作数据可视化大屏主要流程

开通服务

创建项目

配置数据

大屏设计

大屏发布

Step 1 | Step 2 | Step 3 | Step 4 | Step 5

轻松几步配置操作，实现大屏炫酷展示

阿里云

阿里云

机器学习

阿里云大数据分析师ACP认证课程

阿里云

课程目标

学习完本课程后，你将能够：

1. 了解机器学习的基本概念、发展历史、应用场景与基本流程
2. 了解机器学习的常用工具
3. 熟悉机器学习算法的分类与相应的应用场景
4. 了解线性回归、KNN、决策树、Apriori、K-means等算法的原理基本流程

阿里云

阿里云全球培训中心

课程目录

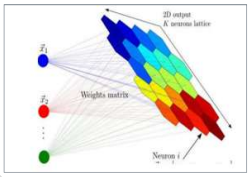
什么是机器学习

机器学习：机器学习是指在不直接针对问题进行编程的情况下，赋予计算机学习能力的一个研究领域

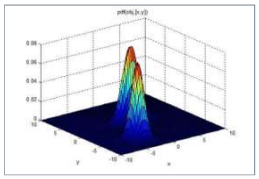
--Arthur Samuel, 1959

机器学习

数据挖掘



模式识别



统计学习

课程目录

机器学习的发展历史

机器学习早期的发展

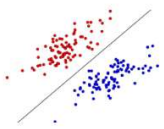
唐纳德·赫布

阿兰·图灵

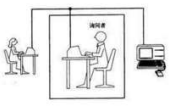
亚瑟·塞缪尔

罗森·布拉特

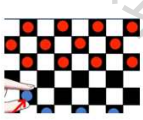
马文·明斯基



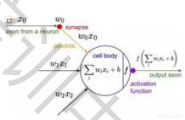
Hebb学习规则



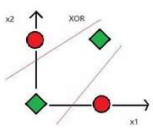
图灵测试



跳棋程序



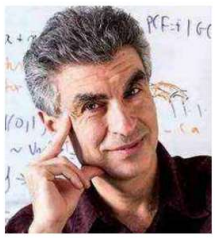
感知机



线性不可分XOR

机器学习的发展历史

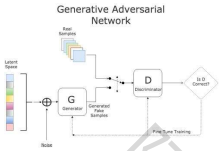
新时代的机器学习



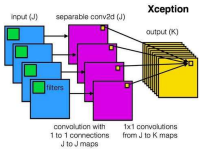
俞书华
Yoshua Bengio

杨乐村
Yann LeCun

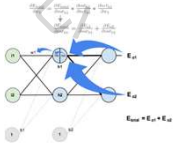
乔福瑞
Geoffrey Hinton



生成对抗器GAN



卷积神经网络CNN



BP反向传播算法

阿里云

课程目录

阿里云

机器学习的应用场景

垃圾邮件检测


信用卡欺诈检测


人脸识别


医疗诊断

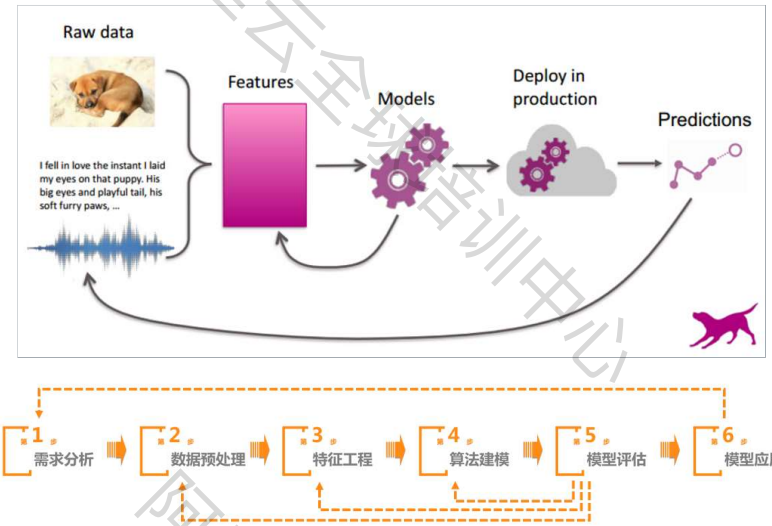

智能客服


无人驾驶

- 专家系统
- 网络信息服务
- 规划与优化
- 图像识别
- 自然语言处理
- 机器人
- 故障诊断
- ……

课程目录

机器学习方法整体流程流程



需求分析

需求分析需要确定机器学习项目的具体需求，包括：

- 场景分析
- 数据收集
- 数据探查
- 算法选择

机场客流量分布预测

为有效利用机场资源，需要提升运行效率的设备包括柜台、商铺、广告位、物流车等。希望通过机场航班信息、wifi数据等信息通过算法实现机场客流量分布猜测

数据预处理 - 数据清洗

数据清洗：针对原始数据，对出现的噪声进行修复、平滑或者剔除，包括但不限于异常值、缺失值、重复记录、错误记录等，同时还会过滤掉不用数据，包括某些行或者某些列。

噪声数据处理：指出现在某变量上的随机误差或变异。简单来说，是指在-组数据中无法解释的数据变动，就是些不和其他数据相一致的数据。

异常值：

- 删除、当做缺失值、忽略
- 分箱法：箱均值、箱中位数或箱边界，对数据进行平滑

缺失值：

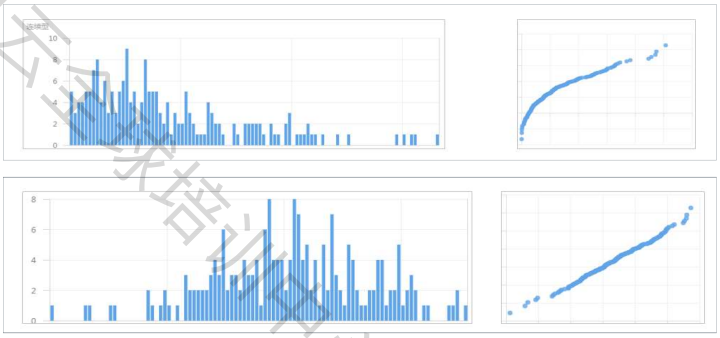
- 统计值填充：均值、众数、中位数等
- 固定值填充：填充指定的值
- 最接近记录值填充：填充和该样本接近的相同字段的值

数据预处理 - 数据变换

数据变换：对数据进行变换处理，使数据更适合当前任务或者算法的需要。

常见的变换方式：

- 使用简单函数进行变换（开根、乘方、对数变换与指数变换等）
- 数据规范化（归一化、标准化、中心化）
- 连续值离散化（分裂法、合并法）



特征工程

特征工程：基于原始数据创建新的特征的过程。一般情况下结合业务，利用数学方法在原有特征的基础上，进行新增、转换等，使特征更适合于机器学习的需要。

特征工程的作用：

- 把原始数据转换成对机器学习更有用的特征
- 结合业务使特征更容易理解和解释
- 结合业务创建对最终结果贡献更大的特征
- 更充分的利用已有数据的价值
- 提供更多的特征，相当于扩充了训练集，充分利用现有的学习、计算能力，提高模型的准确性
- 使用非结构化数据源提供的信息
- 以特征工程的方式引入外部数据

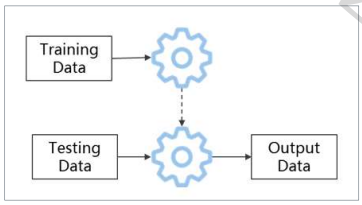
日期	ARPU
201708	180
201707	195
201706	200
201705	240

ARPU高的客户更容易离网?

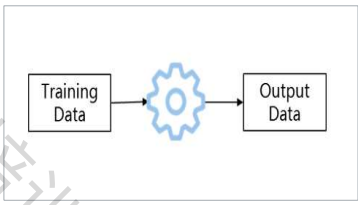
模型训练

模型训练：使用已有的数据输入到选定的模型（算法），调整其参数，使模型的性能和效率可以接受。

有监督学习



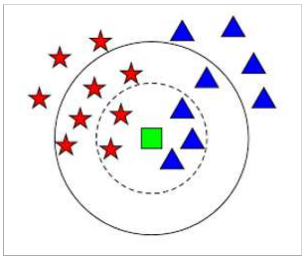
无监督学习



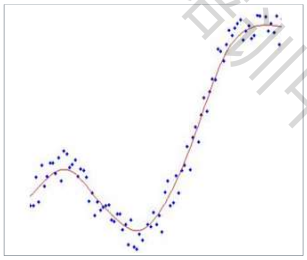
模型评估

模型评估：主要是指从不同维度去评估模型。具体的评价维度依赖于模型的类型和模型的应用场景。

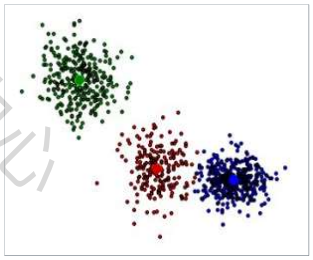
分类模型评估



回归模型评估



聚类模型评估



课程目录

常见开源工具

R语言是用于统计分析、绘图等的操作环境（R Environment），于GNU系统的一个自由、免费、源代码开放的软件，它是一个用于统计计算和统计制图的优秀工具。

Python是种面向对象的解释型计算机程序设计语言，语法简洁清晰，具有丰富和强大的库。常被称为胶水语言，能够把用其他语言制作的各种模块很轻松地联结在-起。

Matlab是一商业数学软件，用于算法开发、数据可视化、数据分析以及数值计算的高级技术计算语言和交互式环境。

Mahout提供了可扩展的机器学习经典算法的实现，更加方便快捷地创建程序。Mahout包含许多实现，包括聚类、分类、推荐过滤、频繁子项挖掘。

课程目录

机器学习PAI

机器学习PAI

阿里云机器学习平台PAI是基于 Maxcompute提供数据处理、建模、离线预测、在线预测等服务的大型机器学习平台，为算法开发者提供了丰富的MPI、PS、BSP等编程框架和数据存取接口，同时提供了基于web的可视化控制台，降低了使用门槛。

上手简单

通过对底层的分布式算法封装，提供拖拉拽的可视化操作环境。让数据挖掘的创建过程像搭积木一样简单。

算法丰富

提供特征工程、数据预处理、统计分析、机器学习、深度学习框架、预测估等超过100种的算法组件

一站式体验

除了提供模型训练功能，还提供了在线预测以及离线调度功能，让机器学习训练结果和业务可以无缝衔接

深度学习

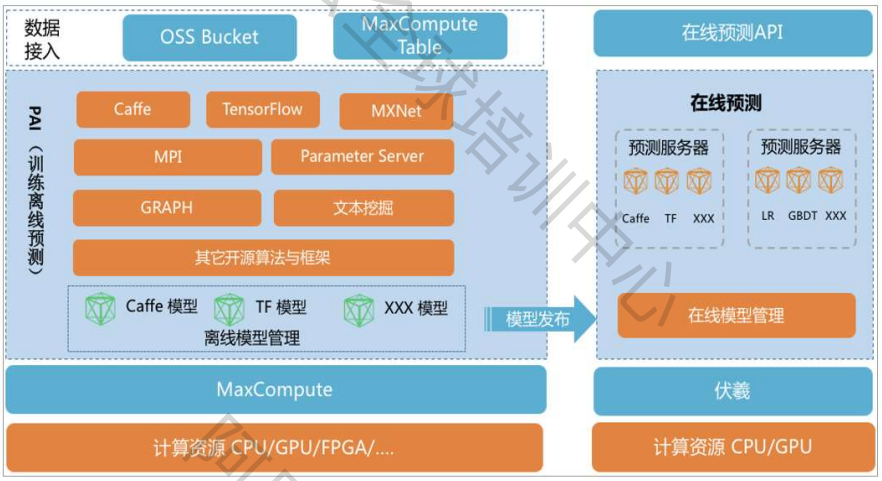
目前支持 MXNet, Caffe, Tensorflow三款主流的深度学习框架，平台底层提供GPU进行训练

机器学习PAI

产品组成

机器学习PAI

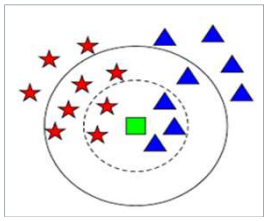
离线模型、在线预测



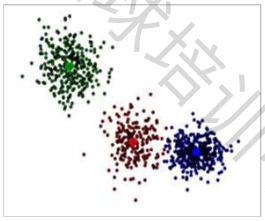
课程目录

常用的机器学习算法分类

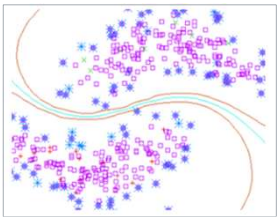
按照**学习方式**不同进行分类：根据学习样本数据不同，对问题有不同的建模方式。



监督式学习
学习样本中有结果标记



无监督学习
学习样本中无结果标记



半监督学习
学习样本中有部分结果标记

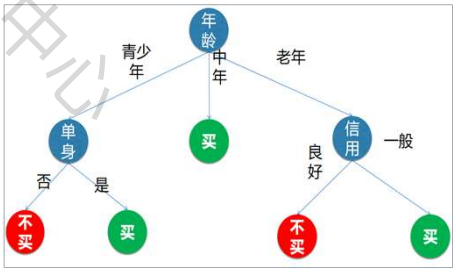
课程目录

分类

分类：就是通过已有数据集（训练集）的学习，得到一个目标函数F（模型），把每个属性集x映射到目标属性y（类），且y必须是离散的（若y为连续的，则属于回归算法）。通过对已知类别训练集的分析，从中发现分类规则，以此预测新数据的类别。

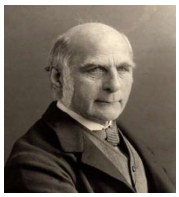
例：已有部分用户是否会购买电脑的记录，据此去建模，来预测某个新用户是否购买。即分成两类：**购买**和**不购买**。

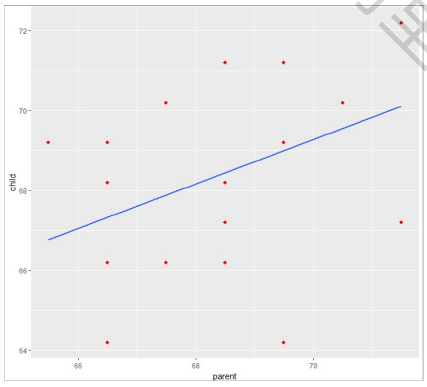
记录标识	年龄	收入层次	是否单身	信用等级	是否购买电脑
1	青少年	高	否	一般	否
2	青少年	高	否	良好	否
3	中年	高	否	一般	是
4	老年	中	否	一般	是
5	老年	低	是	一般	是
6	老年	低	是	良好	否
7	中年	低	是	良好	是
8	青少年	中	否	一般	否
9	青少年	低	是	一般	是
10	老年	中	是	一般	是
11	青少年	中	是	良好	是
12	中年	中	否	良好	是
13	中年	高	是	一般	是
14	老年	中	否	良好	否



回归

回归是处理两个或两个以上变量之间互相依赖的定量关系的一种统计方法和技术，变量之间的关系并非确定的函数关系，通过一定的概率分布来描述。


Francis Galton

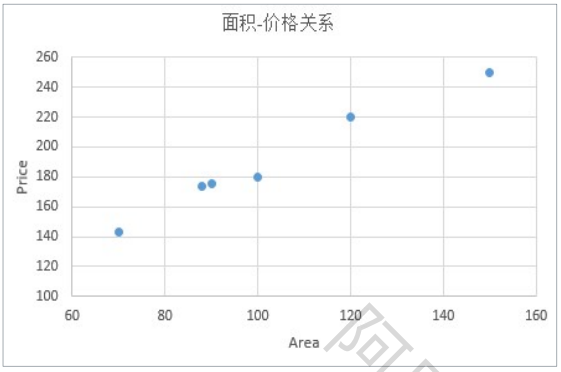


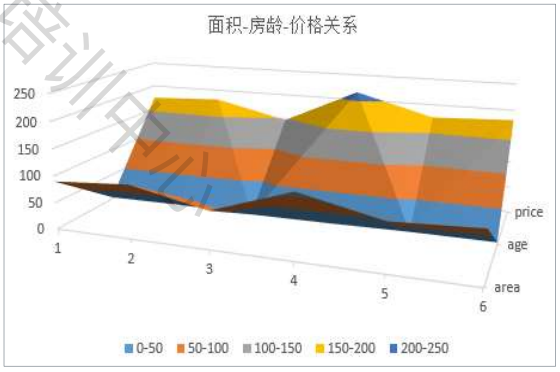
19世纪80年代高尔顿就开始思考父代和子代的相似性问题。搜集了1074对父母及孩子的身高，将结果描成散点图，发现趋势近乎一条直线。

高尔顿对此研究后得出的解释是自然界有一种约束力，使人类身高在一定时期是相对稳定的。如果父母身高（或矮了），其子女比他们更高矮），则人类身材将向高、矮两个极端分化。自然界不这样做，它让身高有一种回归到中心的作用。这个性质被Galton成为“**回归**”。

线性回归

线性回归（Linear Regression）：在回归分析中，如果自变量和因变量之间存在着线性关系，则被称作线性回归。如果只有一个因变量一个自变量，则被称作一元线性回归，如果有一个因变量多个自变量，则被称作多元回归。





回归模型的一般形式

回归模型的一般形式为：$y = f(x_1, x_2, x_3, \dots, x_p) + \varepsilon$

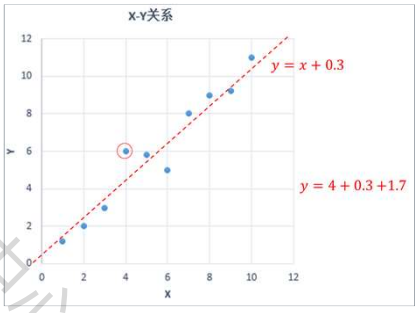
$f(x_1, x_2, x_3, \dots, x_p)$：确定性关系

ε：随机误差（扰动项）
- 影响因素缺失
- 观测/测量误差
- 其他随机误差

当 $f(x_1, x_2, x_3, \dots, x_p)$为线性函数，即：

$$y = f(x_1, x_2, x_3, \dots, x_p) + \varepsilon$$
$$= \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p + \varepsilon$$

$\beta_0, \beta_2, \dots, \beta_p$ 被称作回归系数



一元线性回归

采集了一些人的身高和体重，对这些数据进行建模，然后使用模型，根据输入的身高去预测其体重。

身高	体重
127	25
156	47
160	60
177	93

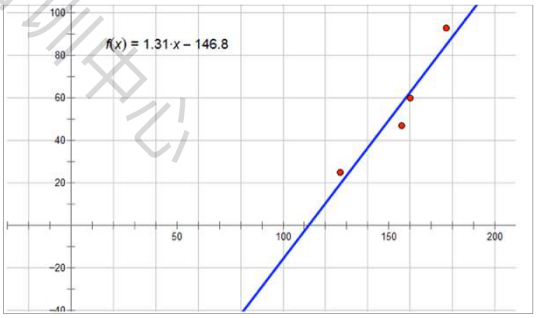
1. 拟合线性方程： Y = a*X + b
2. 带入最小二乘公式：

$$a = \frac{\sum_{i=1}^{n}(h_i - \bar{h})(w_i - \bar{w})}{\sum_{i=1}^{n}(h_i - \bar{h})^2} = 1.31$$

$$b = -146.8$$

3. 模型结果： Y = 1.31*X - 146.8
4. 我的身高180，预测我的体重：

Y = 1.31*180 -146.8 = 89



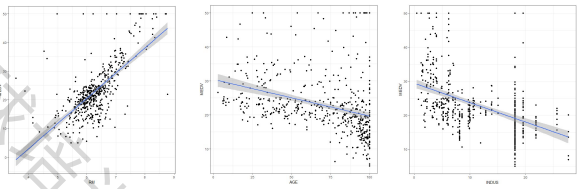
回归模型的特点

回归模型在很多领域都有广泛的应用，具有以下优点：

- 模型简单，建模和应用都比较容易
- 有坚实的统计理论支撑
- 定量分析各变量之间的关系
- 模型预测结果可以通过误差分析精确了解

存在一些缺点：

- 假设条件比较多且相对严格
- 变量选择对模型影响较大



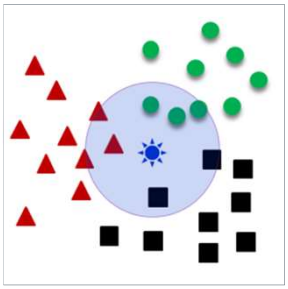
课程目录

KNN（K近邻）

KNN: K-Nearest Neighbour，分类算法中最简单的算法之一，其核心思想是如果离某一个样本最近的k个本中的大多数属于某个类别，则该样本也属于这个类别，并具有这个类别上样本的特性。KNN不但可以预测分类，还可以做回归分析（预测具体的值）。

有N个已知分类结果的样本点，对新记录r使用KNN将其分类的步骤：

- **step1**：确定k值，确定计算距离的公式，比如欧氏距离
- **step2**：计算r和其他样本点之间的距离d_{ir}，其中$i \in (1, N)$
- **step3**：得到目前和r最坡近的k个样本，作为KNN距的训练样本
- **step4**：将k个样本中最多归属类别的分类标签赋予新记录r，分类结束



KNN实现步骤

第一步，确定k的取值，确定距离公式

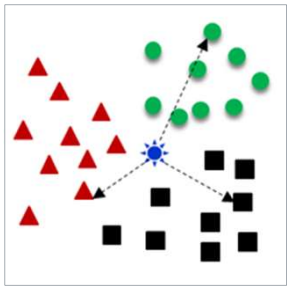
K的取值不宜过大，使用交叉验证确定，本例中选择$k = 10$

有N个已知分类结果的样本点，对新记录r使用KNN将其分类的步骤：

欧氏距离：$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$

曼哈顿距离：$d_{ij} = |x_i - x_j| + |y_i - y_j|$

余弦相似度：$d_{ij} = cos(\overrightarrow{OA}, \overrightarrow{OB})$



KNN实现步骤

第二步，计算r和所有样本点的距离

第三步，圈定符合条件的k条记录

利用公式 $d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$ 计算：

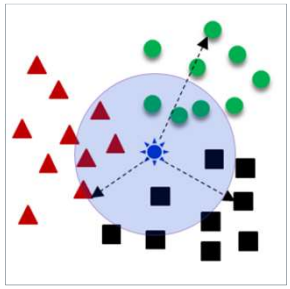
$$d_{1r} = \sqrt{(x_r - x_1)^2 + (y_r - y_1)^2}$$

$$d_{2r} = \sqrt{(x_r - x_2)^2 + (y_r - y_2)^2}$$

$$\cdots\cdots$$

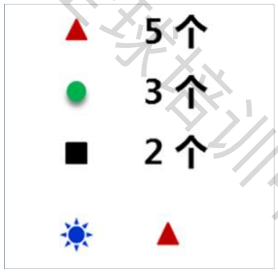
$$d_{Nr} = \sqrt{(x_r - x_N)^2 + (y_r - y_N)^2}$$

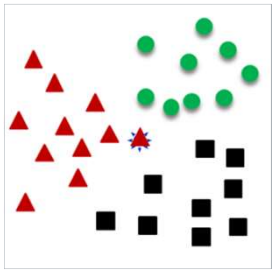
将距离升序排序，得到top k小的值，选入KNN样本



KNN实现步骤

第四步，统计样本分类，确定新纪录r的分类





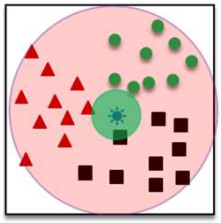
KNN的优缺点

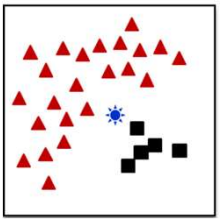
优点：

- 原理简单，容易理解，容易实现
- 重新训练代价较低
- 时间、空间复杂度取决于训练集（一般不算太大）

缺点：

- KNN属于lazy- learning算法，得到结果的及时性差
- k值对结果影响大（试想一下k =1和k = N的极端情况）
- 不同类记录相差较大时容易误判
- 样本点较多时，计算量较大
- 相对于决策树，结果可解释性不强



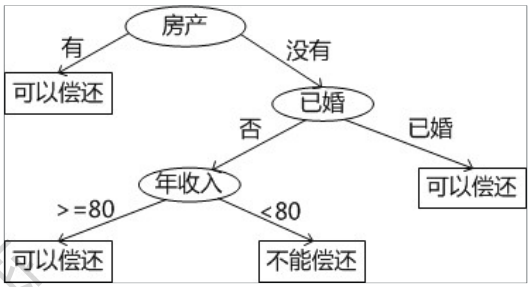


课程目录

决策树的定义

决策树是以事例为基础的归纳学习算法，着眼于从一组无次序，无规则的事例中推导出决策的分类规则。由于这种决策分支图形很像一棵树的枝干，故称决策树。

ID	拥有房产	婚姻状况	年收入	无法偿还债务
1	是	单身	125	否
2	否	已婚	100	否
3	否	单身	90	否
4	是	已婚	120	否
5	否	离婚	75	是
6	否	已婚	60	否
7	是	离婚	220	否
8	否	单身	65	是
9	否	已婚	75	否
10	否	单身	60	是



如果（IF）一个客户，无房，未婚年收入70k，那么（THEN）预测无法偿还债务

决策树的构建流程

准备工作
明确自变量和因变量
确定信息度量的方式
确定终止条件

选择特征
得到当前待处理子集
计算所有特征信息度量
得到当前最佳分类特征

创建分支
根据选中特征将当前记录分成不同分支，分支个数取决于算法

是否终止
判断是否满足终止条件
满足则退出循环
不满足则继续递归调用

生成结果
判断是否需要剪枝
需要则进行适当修剪
不需要则为最终结果

决策树的相关概念

观察数据，明确自变量和因变量

- 自变量
- 因变量

明确信息度量方式：信息增益

- 熵
- 基尼系数

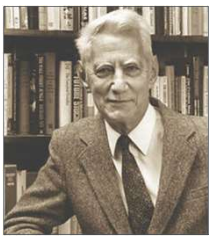
明确分支终止条件

- 纯度
- 记录条数
- 循环次数

根据电脑购买记录，对购买者建模。该模型可以基于客户的一些信息预测他是否会购买电脑。

记录标识	年龄	收入层次	是否单身	信用等级	是否购买电脑
1	青少年	高	否	一般	否
2	青少年	高	否	良好	否
3	中年	高	否	一般	是
4	老年	中	否	一般	是
5	老年	低	是	一般	是
6	老年	低	是	良好	否
7	中年	低	是	良好	是
8	青少年	中	否	一般	否
9	青少年	低	是	一般	是
10	老年	中	是	一般	是
11	青少年	中	是	良好	是
12	中年	中	否	良好	是
13	中年	高	是	一般	是
14	老年	中	否	良好	否

信息熵



信息熵

- 信息论里的概念，香农提出
- 描述混乱程度的度量
- 取值范围0~1，值越大，越混乱
- 计算公式：$H(U)=E[-\log p_i]=-\sum_{i=1}^{n} p_i \log p_i$

有一堆苹果和梨共100个，随机拿出一个，拿出的是哪种？

- 50个苹果+50个梨子 $E1=\left(-\frac{50}{100}\times\log\frac{50}{100}\right)+\left(-\frac{50}{100}\times\log\frac{50}{100}\right)=1$
- 0个苹果+100个梨子 $E2=0+\left(-\frac{100}{100}\times\log\frac{100}{100}\right)=0$
- 80个苹果+20个梨子 $E3=\left(-\frac{80}{100}\times\log\frac{80}{100}\right)+\left(-\frac{20}{100}\times\log\frac{20}{100}\right)=0.722$

信息增益

信息增益

- 信息是确定性的增加
- 从一个状态到另一个状态信息的变化
- 信息增益越大，对确定性贡献越大

名称	颜色	甜不甜	是否为水果
西红柿	红色	否	否
黄瓜	绿色	否	否
苹果	红色	是	是
提子	红色	是	是

不考虑任何特征，直接去估计结果其中，两个是水果，两个不是水果，概率均为$\frac{2}{4}$：

$E=\left(-\frac{2}{4}\times\log\frac{2}{4}\right)+\left(-\frac{2}{4}\times\log\frac{2}{4}\right)=1$

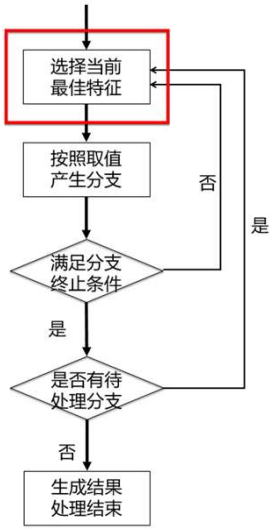
考虑以颜色为参考信息，判断是否为水果：
其中，红色的有3个，2个是水果1个不是水果；绿色的都不是水果。概率分别为$\frac{2}{3}$vs$\frac{1}{3}$与1vs0

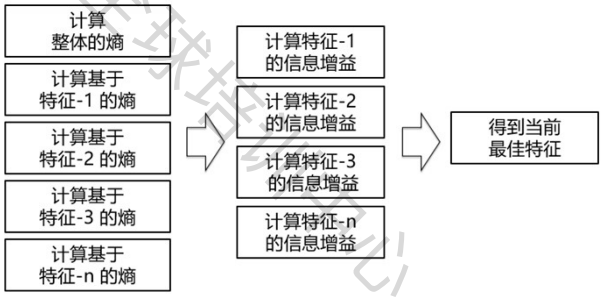
$E=\frac{3}{4}\times\left[\left(-\frac{2}{3}\times\log\frac{2}{3}\right)+\left(-\frac{1}{3}\times\log\frac{1}{3}\right)\right]+\frac{1}{4}\times[(-1\times\log 1)+0]=0.689$

考虑以**味道**为参考信息，判断是否为水果：
其中，甜味的2个，全是水果，不甜的2个，全不是水果，概率分别为：1vs0以及0vs1

$E=\frac{2}{4}\times[(-1\times\log 1)+0]+\frac{2}{4}\times[0+(-1\times\log 1)]=0$

构建决策树





第一级特征选择

计算整体的熵：购买：9例，未购买：5例

$$熵_{整体} = \left(-\frac{5}{14}\times\log\frac{5}{14}\right) + \left(-\frac{9}{14}\times\log\frac{9}{14}\right) = 0.940286$$

计算年龄的熵

年龄	年龄占比	是否购买	购买占比
青	$\frac{5}{14}$	是	2/5
		否	3/5
中	$\frac{4}{14}$	是	4/4
		否	0/4
老	$\frac{5}{14}$	是	3/5
		否	2/5

记录标识	年龄	收入层次	是否单身	信用等级	是否购买电脑
1	青少年	高	否	一般	否
2	青少年	高	否	良好	否
3	中年	高	否	一般	是
4	老年	中	否	一般	是
5	老年	低	是	一般	是
6	老年	低	是	良好	否
7	中年	低	是	良好	是
8	青少年	中	否	一般	否
9	青少年	低	是	一般	是
10	老年	中	是	一般	是
11	青少年	中	是	良好	是
12	中年	中	否	良好	是
13	中年	高	是	一般	是
14	老年	中	否	良好	否

是　　否

$$熵_{年龄} = \frac{5}{14}\times\left[\left(-\frac{3}{5}\times\log\frac{3}{5}\right) + \left(-\frac{2}{5}\times\log\frac{2}{5}\right)\right]_{青}$$
$$+\frac{4}{14}\times\left[\left(-\frac{4}{4}\times\log\frac{4}{4}\right) + 0\right]_{中}$$
$$+\frac{5}{14}\times\left[\left(-\frac{2}{5}\times\log\frac{2}{5}\right) + \left(-\frac{3}{5}\times\log\frac{3}{5}\right)\right]_{老}$$
$$= 0.694$$

第一级特征选择

计算整体的熵：购买：9例，未购买：5例

$$熵_{整体}=\left(-\frac{5}{14}\times\log\frac{5}{14}\right)+\left(-\frac{9}{14}\times\log\frac{9}{14}\right)=0.940286$$

计算收入的熵

$$熵_{收入}=\frac{4}{14}\times\left[\left(-\frac{2}{4}\times\log\frac{2}{4}\right)+\left(-\frac{2}{4}\times\log\frac{2}{4}\right)\right]+\frac{6}{14}\times\left[\left(-\frac{4}{6}\times\log\frac{4}{6}\right)+\left(-\frac{2}{6}\times\log\frac{2}{6}\right)\right]+\frac{4}{14}\times\left[\left(-\frac{3}{4}\times\log\frac{3}{4}\right)+\left(-\frac{1}{4}\times\log\frac{1}{4}\right)\right]=0.911$$

计算单身的熵

$$熵_{单身}=\frac{7}{14}\times\left[\left(-\frac{6}{7}\times\log\frac{6}{7}\right)+\left(-\frac{1}{7}\times\log\frac{1}{7}\right)\right]+\frac{7}{14}\times\left[\left(-\frac{4}{7}\times\log\frac{4}{7}\right)+\left(-\frac{3}{7}\times\log\frac{3}{7}\right)\right]=0.788$$

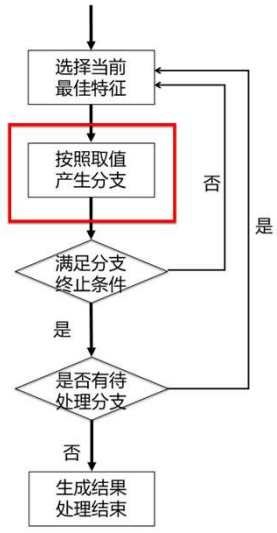
计算信用的熵

$$熵_{信用}=\frac{8}{14}\times\left[\left(-\frac{2}{8}\times\log\frac{2}{8}\right)+\left(-\frac{6}{8}\times\log\frac{6}{8}\right)\right]+\frac{6}{14}\times\left[\left(-\frac{3}{6}\times\log\frac{3}{6}\right)+\left(-\frac{3}{6}\times\log\frac{3}{6}\right)\right]=0.892$$

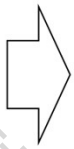
记录标识	年龄	收入层次	是否单身	信用等级	是否购买电脑
1	青少年	高	否	一般	否
2	青少年	高	否	良好	否
3	中年	高	否	一般	是
4	老年	中	否	一般	是
5	老年	低	是	一般	是
6	老年	低	是	良好	否
7	中年	低	是	良好	是
8	青少年	中	否	一般	否
9	青少年	低	是	一般	是
10	老年	中	是	一般	是
11	青少年	中	是	良好	是
12	中年	中	否	良好	是
13	中年	高	是	一般	是
14	老年	中	否	良好	否

特征	特征熵	信息增益
年龄	0.694	0.246
收入	0.911	0.029
单身	0.788	0.152
信用	0.892	0.048

构建决策树



年龄
青少年
青少年
中年
老年
老年
老年
中年
青少年
青少年
老年
青少年
中年
中年
老年

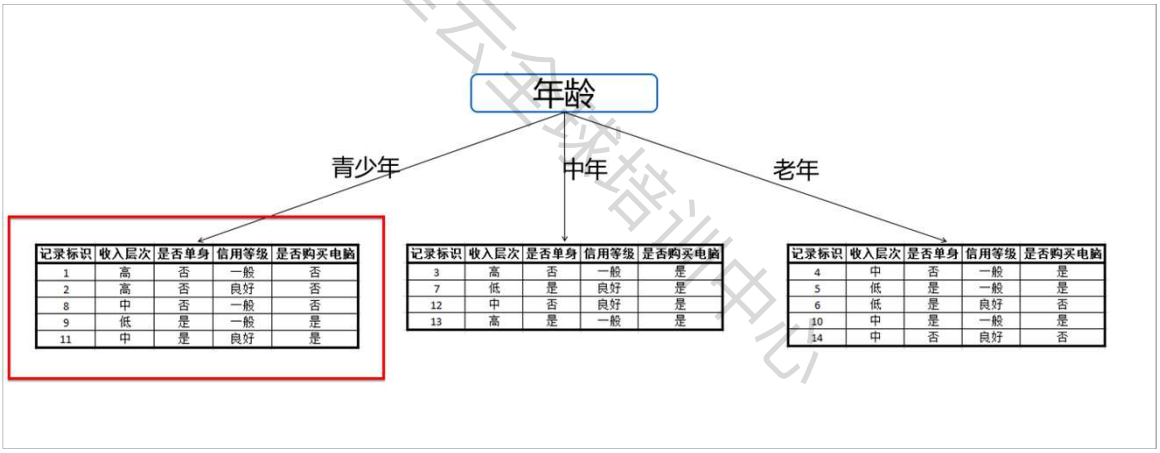


青少年

中年

老年

第一级层分枝

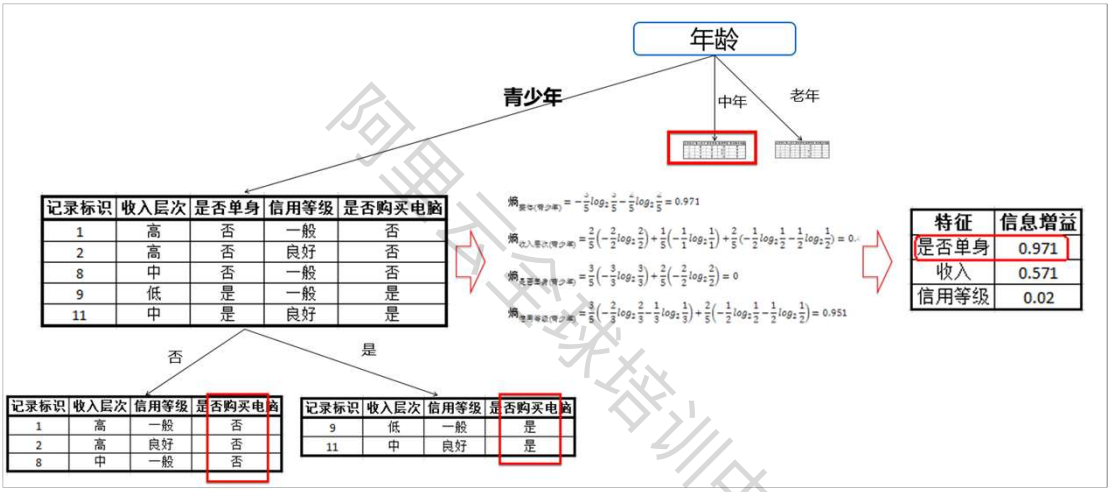


记录标识	收入层次	是否单身	信用等级	是否购买电脑
1	高	否	一般	否
2	高	否	良好	否
8	中	否	一般	否
9	低	是	一般	是
11	中	是	良好	是

记录标识	收入层次	是否单身	信用等级	是否购买电脑
3	高	否	一般	是
7	低	是	良好	是
12	中	否	良好	是
13	高	是	一般	是

记录标识	收入层次	是否单身	信用等级	是否购买电脑
4	中	否	一般	是
5	低	是	一般	是
6	低	是	良好	否
10	中	是	一般	是
14	中	否	良好	否

第二级特征选择



记录标识	收入层次	是否单身	信用等级	是否购买电脑
1	高	否	一般	否
2	高	否	良好	否
8	中	否	一般	否
9	低	是	一般	是
11	中	是	良好	是

$$\text{熵}_{\text{是否购买电脑(青少年)}} = -\frac{2}{5}log_2\frac{2}{5} - \frac{3}{5}log_2\frac{3}{5} = 0.971$$

$$\text{熵}_{\text{收入层次(青少年)}} = \frac{2}{5}\left(-\frac{2}{2}log_2\frac{2}{2}\right) + \frac{1}{5}\left(-\frac{1}{1}log_2\frac{1}{1}\right) + \frac{2}{5}\left(-\frac{1}{2}log_2\frac{1}{2} - \frac{1}{2}log_2\frac{1}{2}\right) = 0.4$$

$$\text{熵}_{\text{是否单身(青少年)}} = \frac{3}{5}\left(-\frac{3}{3}log_2\frac{3}{3}\right) + \frac{2}{5}\left(-\frac{2}{2}log_2\frac{2}{2}\right) = 0$$

$$\text{熵}_{\text{信用等级(青少年)}} = \frac{3}{5}\left(-\frac{2}{3}log_2\frac{2}{3} - \frac{1}{3}log_2\frac{1}{3}\right) + \frac{2}{5}\left(-\frac{1}{2}log_2\frac{1}{2} - \frac{1}{2}log_2\frac{1}{2}\right) = 0.951$$

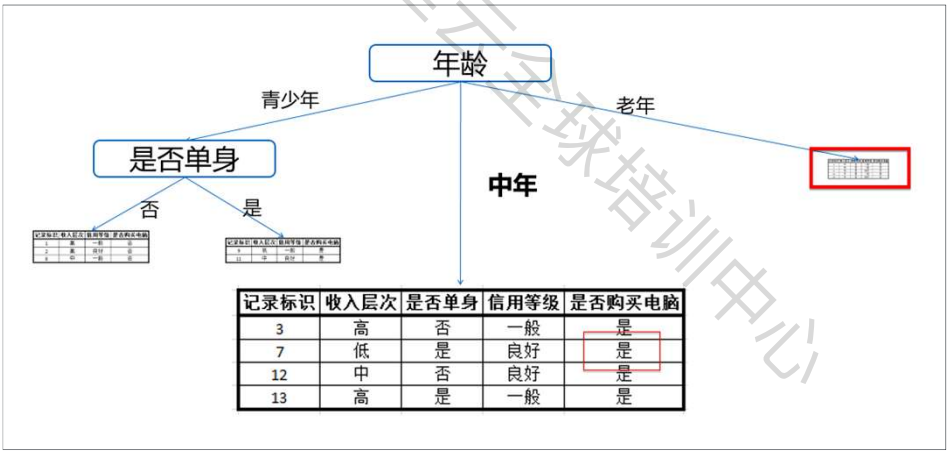
特征	信息增益
是否单身	0.971
收入	0.571
信用等级	0.02

记录标识	收入层次	信用等级	是否购买电脑
1	高	一般	否
2	高	良好	否
8	中	一般	否

记录标识	收入层次	信用等级	是否购买电脑
9	低	一般	是
11	中	良好	是

两个分支的整体熵均为0，纯度已满足退出条件

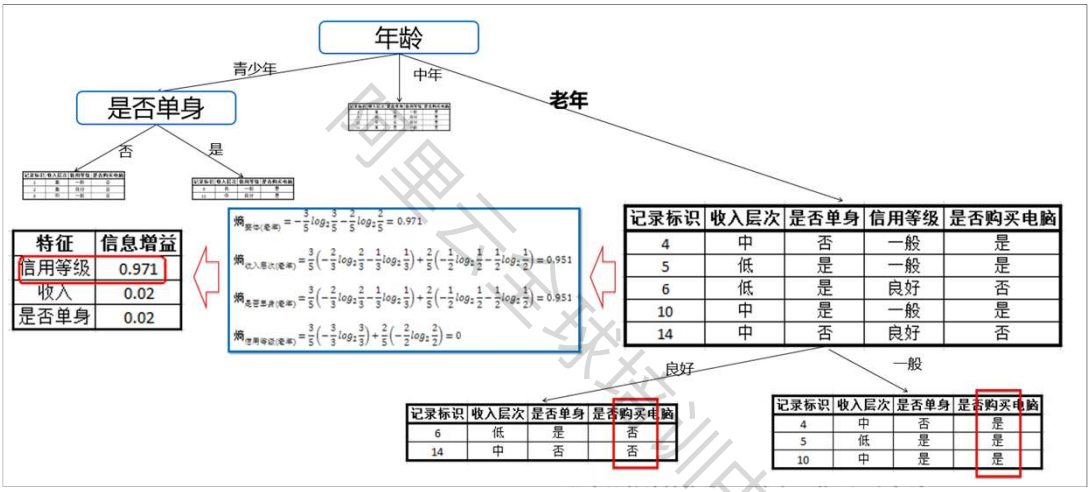
第二级特征选择



记录标识	收入层次	是否单身	信用等级	是否购买电脑
3	高	否	一般	是
7	低	是	良好	是
12	中	否	良好	是
13	高	是	一般	是

整体熵为0，纯度已满足退出条件

第二级特征选择



特征	信息增益
信用等级	0.971
收入	0.02
是否单身	0.02

$$熵_{青少年(是否单身)} = -\frac{3}{5}log_2\frac{3}{5} - \frac{2}{5}log_2\frac{2}{5} = 0.971$$

$$熵_{收入层次(是否单身)} = \frac{3}{5}\left(-\frac{2}{3}log_2\frac{2}{3} - \frac{1}{3}log_2\frac{1}{3}\right) + \frac{2}{5}\left(-\frac{1}{2}log_2\frac{1}{2} - \frac{1}{2}log_2\frac{1}{2}\right) = 0.951$$

$$熵_{是否单身(是否单身)} = \frac{3}{5}\left(-\frac{2}{3}log_2\frac{2}{3} - \frac{1}{3}log_2\frac{1}{3}\right) + \frac{2}{5}\left(-\frac{1}{2}log_2\frac{1}{2} - \frac{1}{2}log_2\frac{1}{2}\right) = 0.951$$

$$熵_{信用等级(是否单身)} = \frac{3}{5}\left(-\frac{3}{3}log_2\frac{3}{3}\right) + \frac{2}{5}\left(-\frac{2}{2}log_2\frac{2}{2}\right) = 0$$

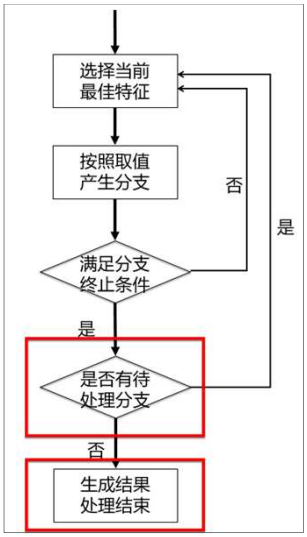
记录标识	收入层次	是否单身	信用等级	是否购买电脑
4	中	否	一般	是
5	低	是	一般	是
6	低	是	良好	否
10	中	是	一般	是
14	中	否	良好	否

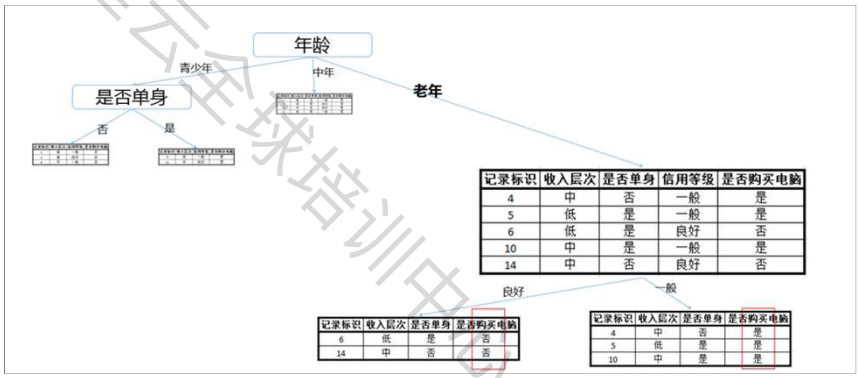
记录标识	收入层次	是否单身	是否购买电脑
6	低	是	否
14	中	否	否

记录标识	收入层次	是否单身	是否购买电脑
4	中	否	是
5	低	是	是
10	中	是	是

两个分支的整体熵均为0，纯度已满足退出条件

构建决策树

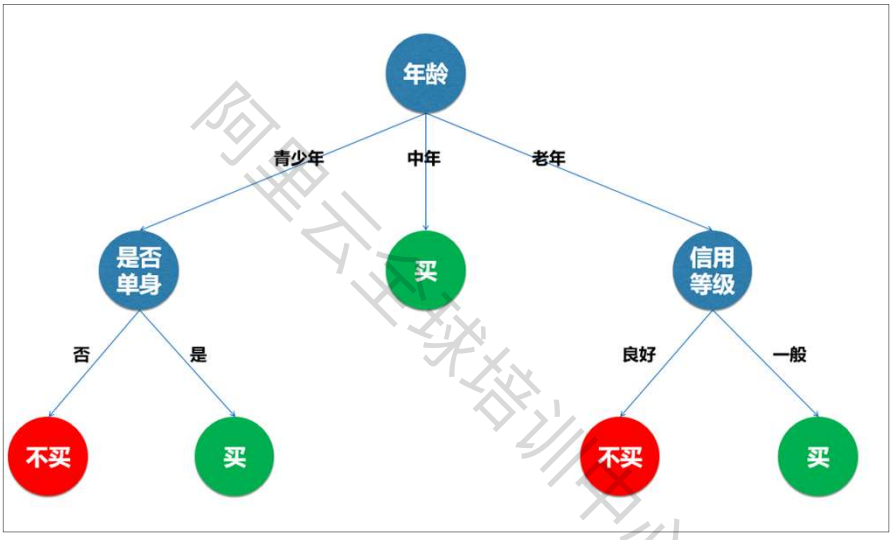




记录标识	收入层次	是否单身	信用等级	是否购买电脑
4	中	否	一般	是
5	低	是	一般	是
6	低	是	良好	否
10	中	是	一般	是
14	中	否	良好	否

两个分支的整体熵均为0，纯度已满足退出条件

生成决策树



ID3系列算法

ID3（Iterative Dichotomise 3，迭代树三代）
- 核心是信息熵，根据信息增益决定树的节点
- 存在以下问题：
  - ✓ 信息度量不合理：倾向于选择取值多的字段
  - ✓ 输入类型单一：离散型
  - ✓ 不做剪枝，容易过拟合

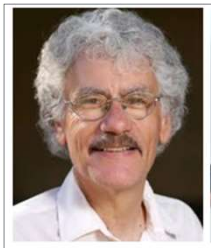
C45：和ID3相比的改进：
- ✓ 用信息增益率代替信息增益
- ✓ 能对连续属性进行离散化，对不完整数据进行处理
- ✓ 进行剪枝

C50：C45相比的改进：
- ✓ 使用了 boosting
- ✓ 前修剪、后修剪

J. Ross Quinlan

分类回归树CART

CART(Classification and Regression Tree)
- 核心是基尼系数(Gini)
- 分类是二叉树
- 支持连续值和离散值
- 后剪枝进行修剪
- 支持回归，可以预测连续值



Jerome Friedman

Leo Breiman

Charles Stone

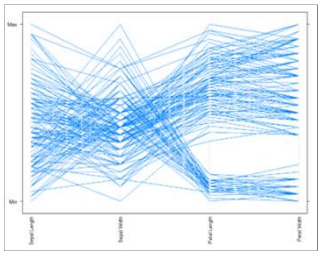
算法	支持模型	树结构	特征选择	连续值处理	缺失值处理	剪枝
ID3	分类	多叉树	信息增益	不支持	不支持	不支持
C4.5	分类	多叉树	信息增益率	支持	支持	支持
CART	分类、回归	二叉树	基尼系数	支持	支持	支持

课程目录

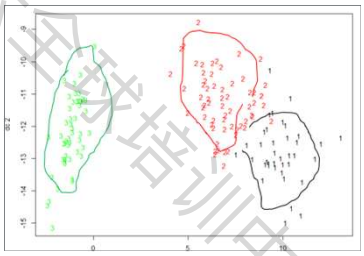
聚类的定义

聚类是指从无标签的数据中，找出某种结构。换言之，将在某些方面比较相似的成员组织到组中，整体数据最终会被分成多个不同的组别。其中，同一个组内的成员相似，不同组内的成员相异。

有未知分类的鸢尾花的测量数据，包括萼长、萼竞、瓣长、瓣竞，根据这些测量数据，将类似的测量记录归类（同种花的测量数据具有类似的特征）。需使用聚类实现“自动”学习。



折线图展示，判断类别数



按照3类聚类后的结果

萼长	萼宽	瓣长	瓣宽
4.6	3.1	1.5	0.2
5	3.6	1.4	0.2
5.4	3.9	1.7	0.4
4.6	3.4	1.4	0.3
5	3.4	1.5	0.2
4.4	2.9	1.4	0.2
4.9	3.1	1.5	0.1
6.2	2.9	4.3	1.3
5.1	2.5	3	1.1
5.7	2.8	4.1	1.3
6.3	3.3	6	2.5
5.8	2.7	5.1	1.9
7.1	3	5.9	2.1
6.3	2.9	5.6	1.8

关联规则的定义

关联规则是反映事物与事物间相互的依存关系和关联性。如果两个或多个事物间存在一定的关联关系则其中一个事物能够通过其他事物预测到。

最常见的场景就是购物篮分析（Market basket）。通过分析顾客购物篮中的不同商品之间的关系，来分析顾客的购买习惯。经典案例就是啤酒与尿布。

课程目录

K-means算法的定义与步骤

K-Means即K均值聚类，属于划分聚类。其工作原理为根据初始化的聚类中心信息，计算每个样本到这些中心的距离，可以判断每个样本均归属于某个类簇，更新聚簇中心信息，重新计算每个样本到新的聚类中心的距离，重新划分样本到新的聚类中心对应的类中，重复进行，直到满足终止条件。

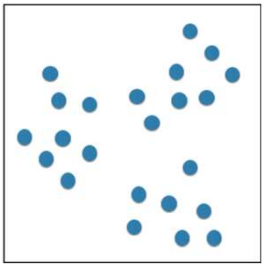
有N个样本点，使用 K-Means将其聚类的步骤：

step1：确定聚类的个数k ，并指定k个聚类的中心$C_1, C_2,\ldots C_k$

step2：计算每个样本S点到k个中心的距离，并将该点归入最近的类中，其中$i \in (1, N), j \in (1, k)$

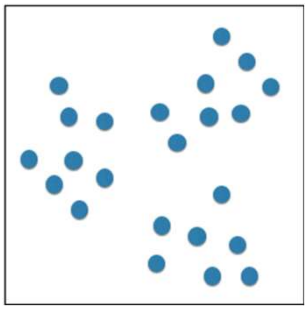
step3：重新计算k个类簇的中心点，更新原有中心点的位置$C_1, C_2,\ldots C_k$

step4：重复步骤Step2、Step3，直到中心点位置不再变化或者变化幅度小于约定阈值，或者达到预定义的最大循环次数，结束。得到最终聚类结果



K-means算法的实现

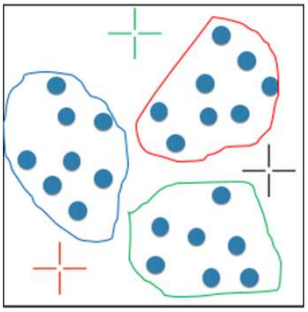
第一步，确定聚类个数，确定聚类中心，确定距离计算公式



观察法

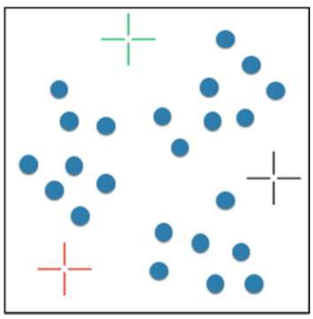
枚举法

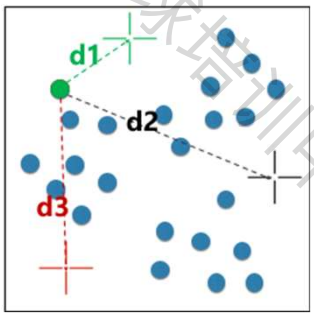
其他技术手段

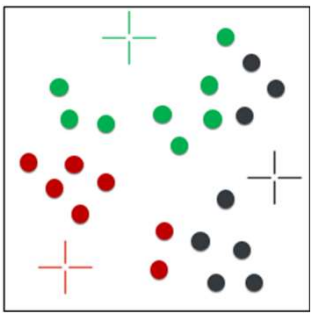


K-means算法的实现

第二步，计算每个点和聚类中心的距离，归类



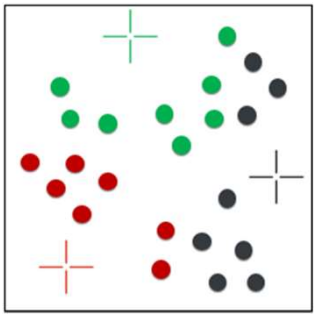


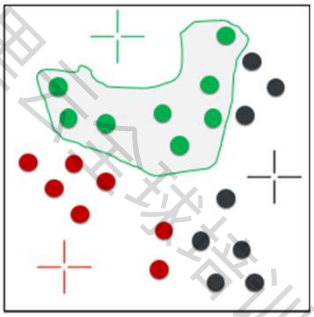


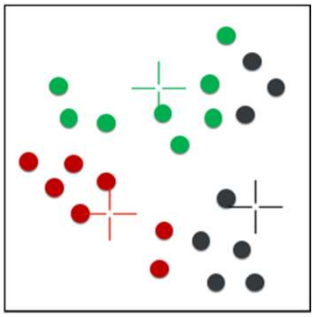
阿里云

K-means算法的实现

第三步，计算当前类簇中心，更新聚类中心C_k的位置





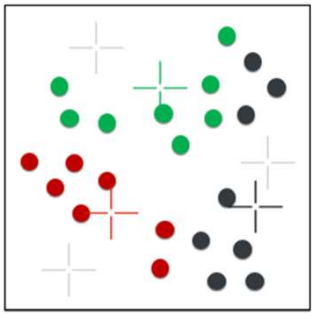
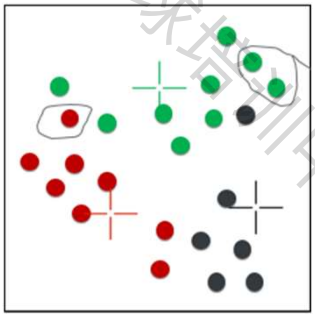
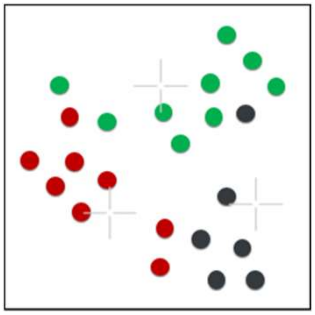


阿里云

K-means算法的实现

重复第二步，将各样本S_i点根据新聚类中心C_k进行重新划分

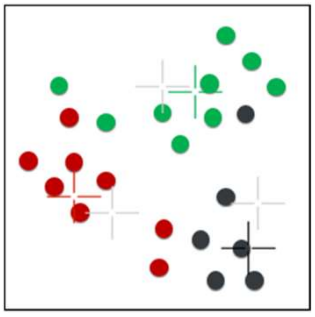
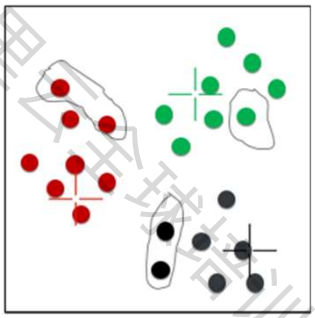
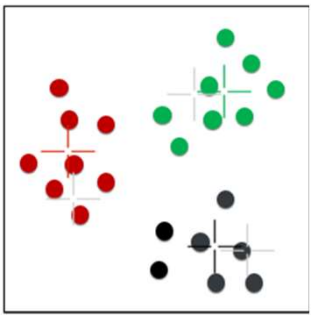
重复第三步，根据最新聚类计算聚类中心，更新中心C_k取值

K-means算法的实现

重复第二步，将各样本S_i点根据新聚类中心C_k进行重新划分

重复第三步，根据最新聚类计算聚类中心，更新中心C_k取值

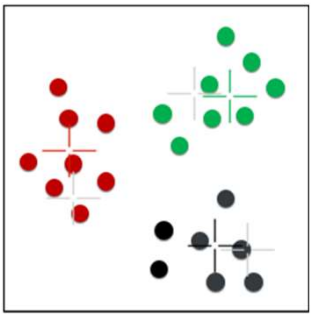
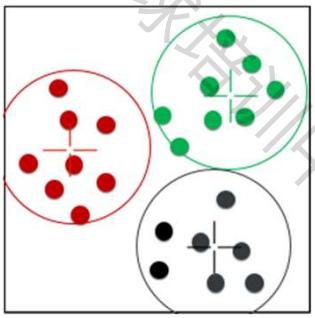
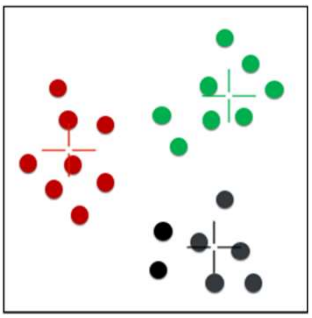




阿里云

阿里云全球培训中心

K-means算法的实现

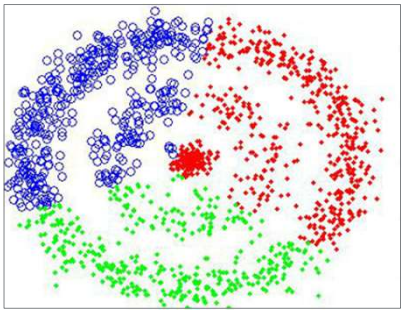
重复第二步，重复第三步，直到聚类中心位置不再发生变化，
或者循环次数达到预先设定的阈值，结束，得到最终聚类结果

K-means算法的优缺点

优点:

- 原理简单，容易理解，容易实现
- 聚类结果容易解释
- 聚类结果相对较好



缺点

- 分类个数k需要事先指定，且指定的k值不同，聚类结果相差较大
- 初始的k个类簇中心对最终结果有影响，选择不同，结果可能会不同
- 能识别的类簇仅为球状，非球状的聚类效果很差
- 样本点较多时，计算量较大
- 对异常值敏感，对离散值需要特殊处理

课程目录

Apriori算法

根据某超市的购物篮信息，分析顾客的购物习惯，制定货物摆放或者捆绑销售策略。（Apriori）

流水号	购买商品
1001	A, C, D
1002	B, C, E
1003	A, B, C, E
1004	B, E

首先确定最小支持度：50%，最小置信度50%

确定1-频繁项集

项集	支持度
{A}	50%
{B}	75%
{C}	75%
{D}	25%
{E}	75%

确定2-频繁项集

Apriori算法

根据某超市的购物篮信息，分析顾客的购物习惯，制定货物摆放或者捆绑销售策略。

流水号	购买商品
1001	A, C, D
1002	B, C, E
1003	A, B, C, E
1004	B, E

首先确定最小支持度：50%，最小置信度50%

确定1-频繁项集

确定2-频繁项集

确定3-频繁项集

项集	支持度
{A, B}	25%
{A, C}	50%
{A, E}	25%
{B, C}	50%
{B, E}	75%
{C, E}	50%

Apriori算法

根据某超市的购物篮信息，分析顾客的购物习惯，制定货物摆放或者捆绑销售策略。

流水号	购买商品
1001	A, C, D
1002	B, C, E
1003	A, B, C, E
1004	B, E

首先确定最小支持度：50%，最小置信度50%

确定1-频繁项集

确定2-频繁项集

确定3-频繁项集

组合	项集	支持度
{A, C} + {B,C}	{A, B, C}	25%
{A, C} + {B, E}	{A, B, C, E}	0%
{A, C} + {C, E}	{A, C, E}	25%
{B, C} + {B, E}	{B, C, E}	50%
{B, C} + {C, E}	{B, C, E}	
{B, E} + {C, E}	{B, C, E}	

Apriori算法

首先确定最小支持度：50%，最小置信度50%

确定1-频繁项集：{A}:50%;{B}:75%;{C}:75%;{E}:75%

确定2-频繁项集: {A,C}:50%;{B,C}:50%;{B,E}:75%;{C,E}:50%

确定3-频繁项集: {B,C,E}:50%

确定关联规则：

流水号	购买商品
1001	A, C, D
1002	B, C, E
1003	A, B, C, E
1004	B, E

规则	支持度	置信度
B -> C	50%	66.7%
B -> E	75%	100%
B -> CE	50%	66.7%
C -> E	50%	66.7%
CE -> B	50%	100%
……	……	……

数据分析报告撰写

阿里云大数据分析师ACP认证课程

阿里云

课程目标

学习完本课程后，你将能够：

1. 了解数据分析报告的撰写的**原则、种类、作用及结构**
2. 能**撰写**数据分析报告
3. 能针对数据分析报告进行**讲解**

阿里云

课程目录

1. **初识数据分析分析报告**
2. 数据分析报告的种类
3. 数据分析报告的结构
4. 数据分析报告撰写注意事项

阿里云

什么是数据分析报告

- 数据分析报告是根据数据分析原理和方法，运用数据来**反应、研究和分析**某项**事物现状、问题、原因、本质和规律**，并得出结论，**提出解决问题办法**的一种分析应用文体。

阿里云

数据分析报告的作用

- 能够迅速理解、分析、研究问题的基本情况、结论与建议等内容

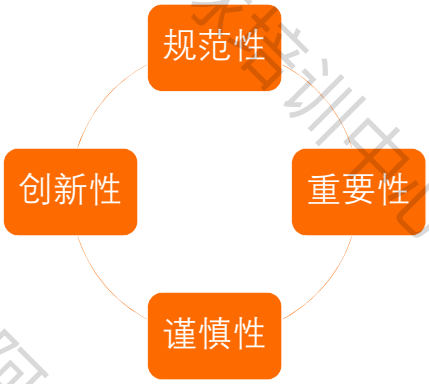
- 总结、检验数据分析质量，增加分析严谨性
- 具有时效性，决策方面的重要参考依据

数据分析报告的目标

1. 确定项目重点
2. 进行总体分析
3. 总结经验

数据分析报告的撰写原则

- 一份完整的数据分析报告，应当**围绕目标确定范围**，遵循一定的前提和原则，系统地**反映存在的问题及原因**，从而进一步找出解决问题的方法。需要遵循以下4个原则。



数据分析报告的撰写原则-规范性

- **规范性**：数据报告要“以数据说话”。
  - 所使用**数据单位、名词术语一定要标准统一，前后一致**，基本上要与前人所提出的相一致。
  - 所使用指标的**数据来源要有清晰**的说明，从数据管理系统采集的，要说明系统名称。
  - 现场测量的要说明抽样方式、抽样量和测量时间段等。

数据分析报告的撰写原则-重要性

- **重要性**：数据分析报告的输出是整个分析过程的成果，是评定一个产品、一个运营事件的**定性结论**，很可能是产品**决策的参考依据**。因此数据分析报告一定要体现项目分析的重点，在项目各项数据分析中，就应该重点选取**真实性、合法性指标，构建相关模型，科学专业**地进行分析，并且反映在分析结果中对同一类问题的描述中，也要按照问题的重要性来排序。
  - 在结构上，数据分析报告应根据项目目标，对确定优化事项进行**重点分析，详尽展示，对优化事项**可能造成的边缘性影响进行**辅助说明，条理清晰，重点突出。**
  - 在内容上，对同一类问题分析描述时，要按照问题的**重要性分级**，重要在前，次重要在后。

数据分析报告的撰写原则-谨慎性

- **谨慎性**：数据分析报告的编制过程一定要谨慎。
  - 基础数据须要**真实完整。**
  - 分析过程须要**科学合理全面。**
  - 分析结果可靠，建议内容**实事求是。**

数据分析报告的撰写原则-创新性

- **创新性**：科技发展、进步，创新的方法或模型从实践中摸索总结出来，数据分析报告要将这些创新的想法记录下来，发扬光大。
  - 引进一些新的**分析方法和研究模型**，在确保数据真实的基础上，提高数据分析的**多样性。**
  - 倡导**创新性思维**，提出的优化建议在考虑企业实际情况的基础上，要有一定的**前瞻性、操作性、预见性。**

课程目录

1. 初识数据分析分析报告
2. **数据分析报告的种类**
3. 数据分析报告的结构
4. 数据分析报告撰写注意事项

数据分析报告的种类

- 由于数据分析报告的**对象、内容、时间、方法**等情况的不同，因而存在着不同形式的报告类型。我们常用的几种数据分析报告有日常数据通报、综合分析报告、专项分析报告等。

数据分析报告的种类-日常数据通报

- **日常数据通报**：以定期数据分析报表为依据，**反映计划**的执行情况，并**分析其影响和形成原因**的一种数据分析报告。
- **适用场景**：对**绩效结果的展示**，具有导向性，对决策修订或管理改进具有指导意义。
- **报告特点**：
  - 进度性
  - 规范性
  - 时效性

数据分析报告的种类-综合分析报告

- **综合分析报告：**又叫全面分析报告。它是指某一部门、某一单位或某地区把一定时期的经济活动作为一个整体，对各项主要经济指标的完成情况进行**综合分析研究**，从中找出带有普遍性和关键性的问题，认识其规律，以图改进的报告文书。
- **适用场景：**既可用于**宏观分析**，也可用作**微观分析**，是全面评价一个地区、单位、部门业务或其他方面发展情况的一种数据分析报告。
- **报告特点：**

全面性

联系性

数据分析报告的种类-专项分析报告

- **专项分析报告**：又叫专题分析报告，它是指某一部门、某一单位或某地区把某活动化整为零，对某项或某类经济指标进行分析研究，找出规律并提出方案。此报告**内容单一**，但**分析更加深入。**
- **适用场景：**为决策者制定某项政策、解决某个问题提供决策参考和依据。
- **报告特点：**

单一性

深入性

数据分析报告的种类展现形式

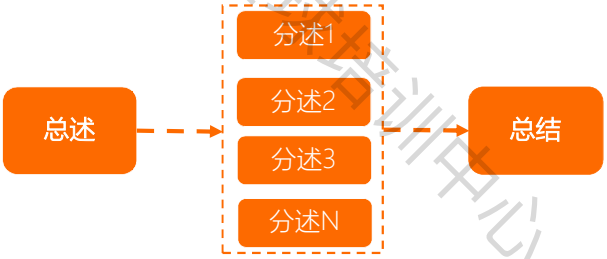
- 数据分析报告的展现形式主要通过办公软件中的**Word、Excel和PowerPoint**来表现。这三种软件**各有优劣势**，具体对比信息和适合分析报告应用场景范围如下：

优势	• 易于排版 • 可打印装订成册	• 含有动态图表 • 结果可实时更新 • 交互性更强	• 可加入丰富的元素 • 适合演示汇报 • 增强展示效果
劣势	• 缺乏交互性 • 不适合演示汇报	• 不适合演示汇报	• 不适合大篇文字
使用范围	• 综合分析报告 • 专题分析报告 • 日常数据通报	• 日常数据通报	• 综合分析报告 • 专题分析报告

课程目录

数据分析报告的结构

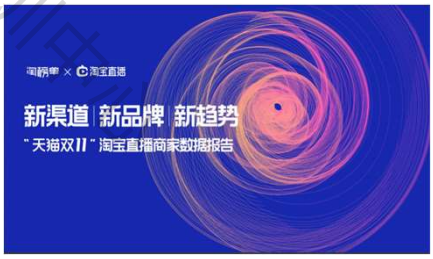
- 最经典的报告结构还是“**总—分—总**”结构，它主要包括：**开篇、正文和结尾**三大部分。



“总-分-总”结构详细介绍

数据分析报告的结构-标题页

- 标题页需要写明报告的题目，题目要精简干练，根据版面的要求在一两行内完成。标题是一种语言艺术，好的标题**新鲜活泼、独居特色**，不仅可以表现数据分析的主题，而且能够激发读者的阅读兴趣，因此需要重视标题的制作，以增强其艺术性的表现力。标题制作要求：**直接、确切、简洁。**



数据分析报告的结构-目录页

- 目录是报告的大纲、体现报告的分析思路。

简洁

概括性

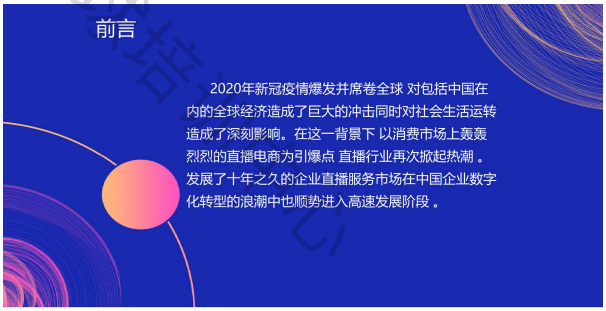
数据分析报告的结构-前言

- 前言分析报告背景和目的。

分析背景

分析目的

分析思路



数据分析报告的结构-正文

- 正文是数据分析报告的核心部分，它将系统全面地**表述数据分析的过程与结果。**

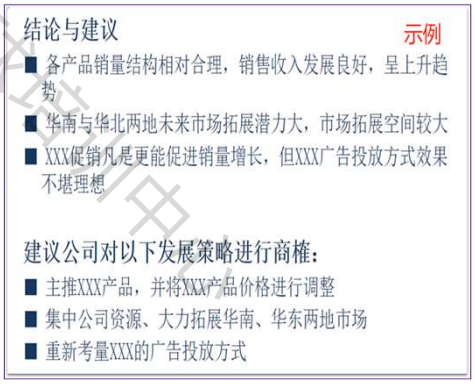
数据分析报告的结构-结论与建议

综合、总结

深化、提高

引发思考

严谨、准确、鲜明



数据分析报告的结构-附录

- 附录是数据分析报告的一个重要组成部分。附录提供正文中涉及而**未予阐述的有关资料**，有时也含有正文中提及的资料，从而向读者提供一条深入数据分析报告的途径。它主要包括报告中涉及的专业名词解释、计算方法、重要原始数据、地图等内容。每个内容都需要编号，以备查询。
- 附录是**数据分析报告的补充，并不是必需的**，应该**根据各自的情况再决定**是否需要在报告结尾处添加附录。

课程目录

1. 初识数据分析分析报告
2. 数据分析报告的种类
3. 数据分析报告的结构
4. **数据分析报告撰写注意事项**

阿里云

数据分析报告撰写注意事项

**结构合理
逻辑清晰**

- 数据分析报告的**结构是否合理、逻辑条理是否清晰**是决定此份报告成败的关键因素。

**实事求是
反映真相**

- 数据分析报告**最重要**的就是必须具备真实性。

**用词准确
避免含糊**

- 尽量**用数据说话**，避免使用“大约”“估计”“更多”“更少”等**模糊字眼**。

**篇幅适宜
简洁有效**

- 篇幅长的报告不一定是好的报告!

**结合业务
分析合理**

- 分析报告不只是基于数据而分析问题，或简单地看图说话，**必须紧密结合具体业务**才能得出可实行、可操作的建议，否则将是**纸上谈兵，脱离实际。**

阿里云

知识点回顾

数据分析报告

- 数据分析报告**定义、作用、目标**
- 数据分析报告的撰写原则：**规范性、重要性、谨慎性、创新性**
- 数据分析报告的三种类型：**综合分析报告、专项分析报告、日常数据通报**
- 数据分析报告的结构：**标题、目录、前言（分析目的与背景分析结论）、正文、建议、附录**
- 数据分析报告撰写注意事项

数据分析项目管理

阿里云大数据分析师ACP认证课程

阿里云

课程目标

学习完本课程后，你将能够：

1. 掌握项目、项目管理、项目管理方法论的概念
2. 掌握数据分析项目的实施流程
3. 掌握经常型项目与临时型项目概念、区别
4. 掌握项目前分析、项目后分析作用
5. 掌握数据项目设计的方法

阿里云

阿里云全球培训中心

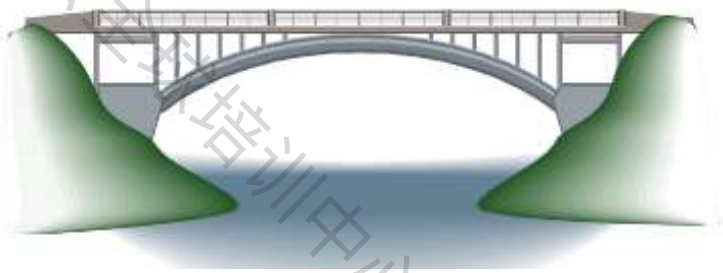
课程目录

项目的定义

项目是为创造独特的产品、服务或成果而进行的临时性努力。

它具有如下相关特性：

- 临时性(Temporary)
- 独特性(Unique)
- 渐进明细(Progressive Elaborat



哪些是项目

- 一家公司被收购
- 海外建设通讯站点
- 研究院研发一种新型的药物
- 修建一座跨海大桥
- 野生动物湿地保护区的建立
- 与朋友一起旅游
- 房屋装修
- 公司车间产品产能不足分析

某食品公司新研发出一种休闲的食品，独立小包装为12g，每箱放置30包独立小包装，近期销售较好，出现供不应求的势头。
产品经过车间生产出来，在包装入箱后，存入成品库时，却多批次的被检验人员给退回车间进行返工，原因为频繁的出现部分批次产品的包装重量显著性的不够达标。
通过车间产品的流程的数据，分析数据，找出包装产品产能不达标的真正原因。

- **……请问哪些是项目?**

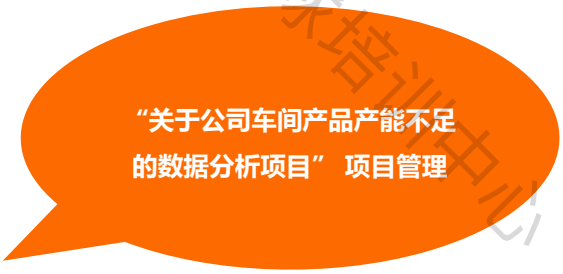
什么是成功的项目

项目成功的定义-三要素

- 按时完成
- 预算内
- 质量符合预期要求:
  - 功能
  - 性能

项目管理

项目管理是把各种知识、技能、工具和技术应用于项目各项活动中，以达到项目的要求，完成项目的目标。



阿里云

项目管理方法论

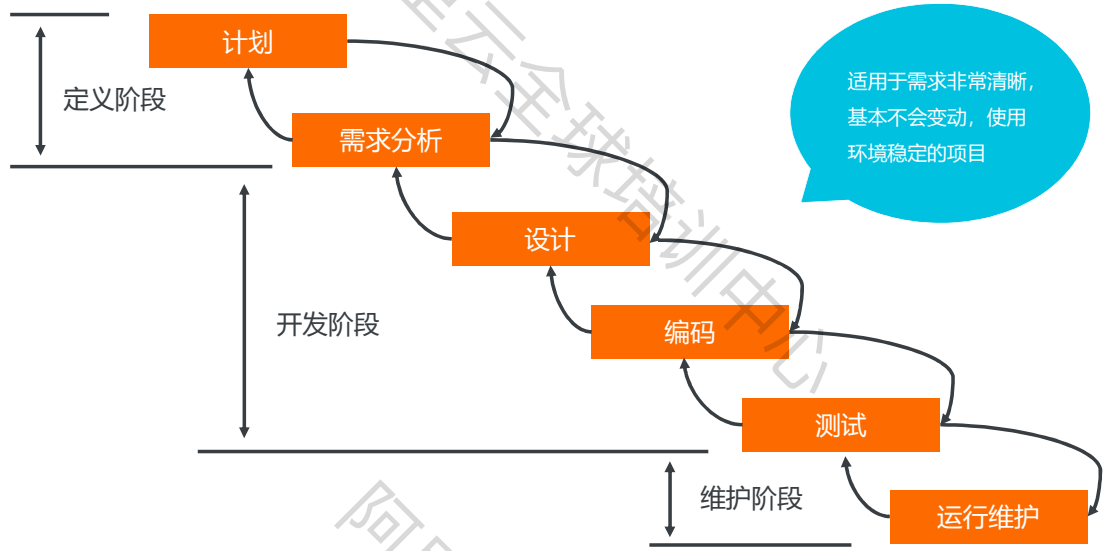
阿里云

项目管理方法论-PMBOK

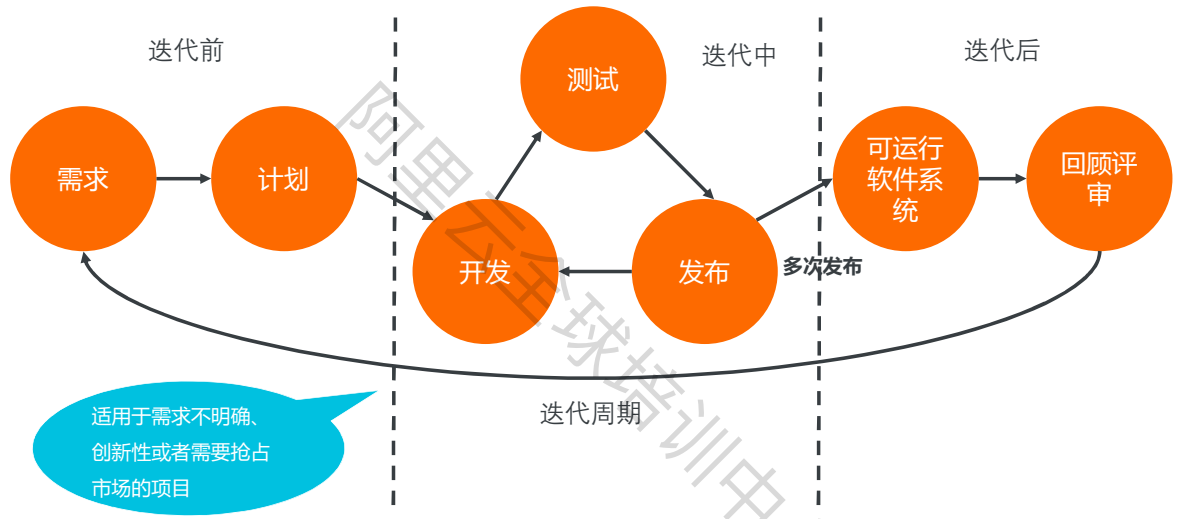
项目管理方法论-PRINCE2

七个项目管理原则	持续业务验证	经验学习	角色与责任	按阶段管理	例外管理	关注产品	剪裁
七个项目管理主题	业务论证	组织	质量	计划	风险	变更	进展
七个项目管理流程	项目指导	项目发起	项目准备	项目阶段边界管理	项目阶段控制	项目产品交付	项目收尾
四层项目管理组织结构	公司战略管理层	项目委员会	项目经理	项目小组经理			
二个项目管理技术	产品计划：在定义活动之前先识别和定义产品	质量评审：定义了应用到所有产品的质量控制和审批过程					

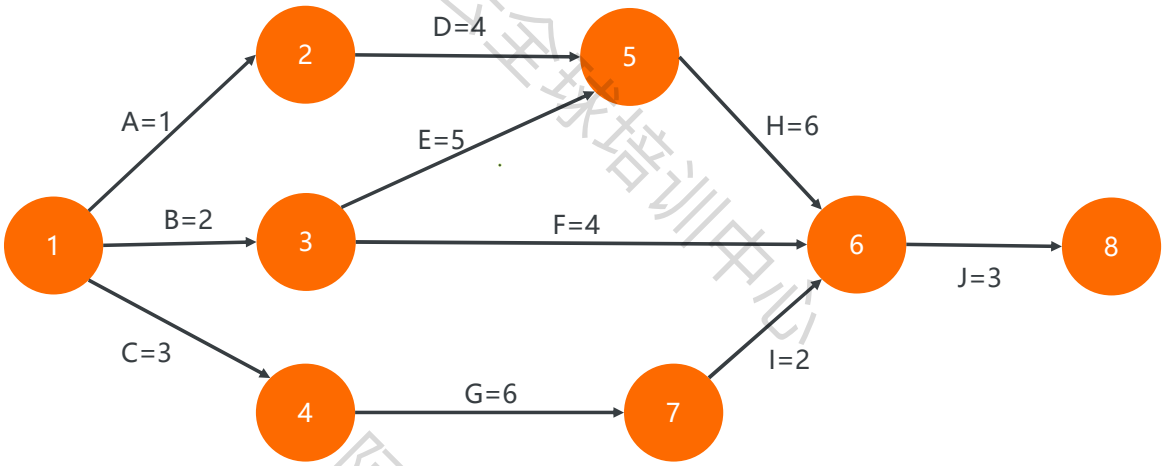
项目管理方法论-瀑布开发



项目管理方法论-敏捷开发



项目管理方法论-关键路径



项目管理方法论-其它

- **混合模型**：充分发挥瀑布模型的严谨、以及敏捷开发速度快等优点，计划和需求阶段是在瀑布方法下进行的，而设计、开发、实现和评估阶段则遵循敏捷方法。
- **看板管理**：使用可视板和卡片来帮助团队查看已完成、正在进行和未完成的任务。所有的活动都是基于对日常任务的可视化、对工作进度的仔细平衡和对待办事项的管理。
- **六西格玛**：主要强调制定极高的目标、收集数据以及分析结果，通过这些来减少产品和服务的缺陷。
- **关键链项目管理（CCPM)**：被用来替代关键路径分析方法（CPM）。关键链项目管理与关键路径法的不同之处在于，它关注项目内部资源的使用，而不是项目活动。为了解决资源的潜在问题，内置了缓冲区，以确保项目按时进行，并且不会损害安全性。关键链项目管理使用缓冲管理而不是挣值管理来评估项目的绩效。。

课程目录

阿里云

数据项目

- 数据项目主要分两种：大数据工程项目、数据分析项目
- 两者有些技能通用，但使用的工具有所不同。
- 两者项目流程相似，但具体的实施差别较大。

大数据工程项目

- 包括系统搭建和数据采集、处理、入库等。
- 需要较多的IT技能。

数据分析项目

- 包括业务报表、洞察分析、统计数据挖掘建模。
- 需要较多的分析技能。

阿里云

数据分析项目分类

Routine Job
经常型项目

Ad hoc
临时型项目

两种项目的区别在于项目的设计和实施对编程要求不一样

阿里云

经常型项目

经常型项目通常需要定期的执行（一些小的数据项目会要求工作人员手动定期执行）

定期执行

自动调度

项目程序会被要求让系统按预先设定的时间自动运行

执行程序过程中耗费的时间需要降到最低，即提高程序的运行效率。

关注程序运行效率

代码固定

反复使用同一程序执行

提高程序的重复利用率，使用大量的宏变量，将手动修改程序执行的情况降到零

减少手工调整操作

全自动化

提高程序输出结果的自动化程度，让程序直接输出报表，甚至直接发送到报表的使用者手里

阿里云

临时型项目

新项目

随时可出现

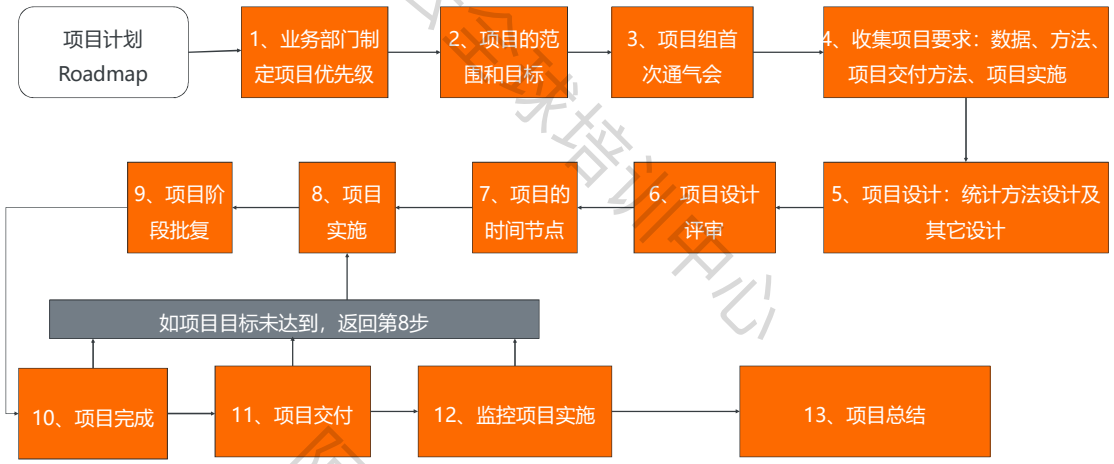
遵循标准的项目设计和实施的流程

关注质量

关注紧急程度

课程目录

数据分析项目实施流程



项目计划阶段

- 在项目计划阶段，主要由业务部门制定项目的计划列表。业务部门根据企业的愿景目标制定数据项目。
- 大致流程如下：

- 在项目计划（roadmap）阶段，数据项目的执行流程中的步骤1（项目优先级）和步骤2（项目的范围和目标）也同期完成。

项目前分析

- 项目立项前的分析是对项目开展的原因以及在考虑到项目成本估算、风险及收益预测等因素的情况下，项目可行性和优化提高机会的分析。
- 分析主要解决：

项目是否可以顺利执行，如目标客户是否足够多，市场容量是否足够大?

项目在实施过程中是否有优化提高的空间，如何提高等?

项目的投入成本和预计的产出?

实验设计，保证项目后续效考评能够顺利执行。

项目计划

- 针对车间产能不足的情况，食品公司进行项目立项，明确项目目标、时间、预算。
- 食品公司业务部门对项目进行分析，明确初始的项目目标及范围，并确定项目的优先级。
- 业务部门编写数据项目需求说明书，明确数据项目范围，避免需求蔓延。
- 将项目分解为更多的工作细目或者子项目，使项目变得更小、更易管理、更易操作
- 业务部门进行任务分解、成本估算、资源估算、进度安排；
- 估算项目需要的工作量，和完成的时间节点，要求此项目1个月内完成。

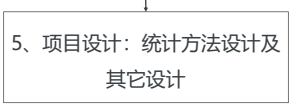
项目首次通气会

- 项目的管理者会根据项目实施的时间，提前一段时间召开全体项目人员的会议。
- 对项目的整体情况作介绍，目的是让有关人员对项目的具体情况达成共识。
- 并且对参与人员的工作量予以初步估算，收集信息以帮助安排项目分解后各个子项目的时间节点。

- 数据分析师，在该会议中，有机会提出想关的问题，例如：
  - 1：项目设计中涉及的数据可能不存在，需要调整。
  - 2：项目设计的范围可能太大，需要调整。
  - 3：项目设计的分析人员需要调整等等

理解项目需求

- 分析需求，收集项目要求



数据分析人员进行了调查，排除了人员问题。已知该产品的生产共有4条生产线，每条生产线配置了一台包装机自动进行包装，包装的标准克重是12g/包，同时包装机的使用也有一些年限了。

- 包装机有没有出问题，如果出了问题，是哪些包装机?
- 包装机的速度和包装的大小有关系吗？如果有关系，那最佳速度是多少?
- 为保证数据样本多样性，在一天的不同时刻对4台包装机同时抽取了样本容量为100的包装数据，同时记录下了当时包装机速度。
- 最终提供一份数据分析报告，找出具体是哪台包装机出现问题。

项目设计

- 充分理解项目的要求后，对具体的任务设计具体实施的方法。
- 这些方法可能指的是：

语言和工具	分析方法	数据源需求
如SQL，Python，或者是Java，SAS等等。	或者是统计模型、数据挖掘算法、人工智能等等	大多数统计模型对脏数据比较敏感。而部分高级算法对数据属性的数量有要求。数据源的问题同时也会逼迫分析师选择不同的数据处理或分析方法。

- 在项目设计阶段，数据项目的执行流程中的步骤6（项目设计peer review）和步骤7（项目的时间节点）也同期完成。

项目设计

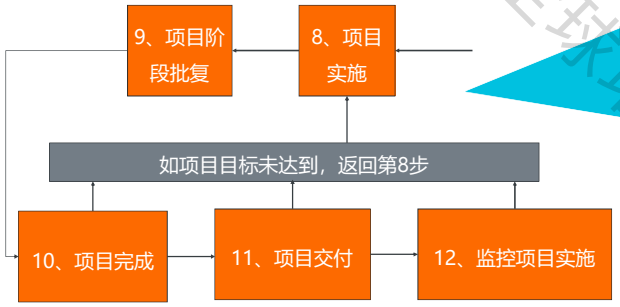
5、项目设计：统计方法设计及其它设计 → 6、项目设计评审 → 7、项目的时间节点

设计是否覆盖需求，算法及工具选择是否合理？

- 统计分析
  - 通过统计分析，看各样本均值与标准的大小来确定那台设备有问题：
  - T检验、假设检验
- 线性回归
  - 特征列为速度，标签列为包装大小，通过线性回归算法，预算包装机的速度。
- 高效性：在设计过程中，充分考虑数据量，使设计满足不同量的数据；
- 准确性：在设计过程中，选择适当的方法及工具，找到真正的项目原因；
- 工具：使用阿里云机器学习PAI，大数据计算服务MaxCompute

- 确定项目开发的时间节点（重要里程碑）
- 例如：
  - 数据采集至MaxCompute
  - 数据预处理（清洗）完成
  - 使用PAI完成程序开发
  - 数据统计分析报告完成

项目实施



- 项目实施阶段主要根据前面项目设计的结果、步骤进行开发实施；
- 在每个重要时间节点完成各个里程碑任务；
- 同时每个里程碑阶段进行会议沟通汇报（汇报进度、遇到的问题及解决方案等）；
- 如分析项目在整体控制内，在规定的时间内完成，则将最终的分析结果交付给需求方。
- 项目执行的过程中，每一个阶段都需要进行一定的监控，保证项目少走弯路，按时按质的完成。

阿里云

项目实施

- 数据采集至MaxCompute中，查看源数据统计情况

  看下每台包装机包装产品的样本均值，目的是看各样本均值与标准的大小来确定原假设及备择假设：

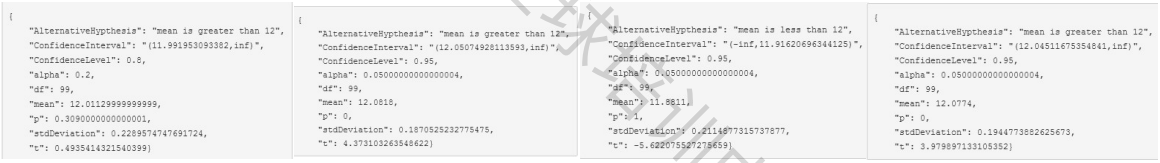
min	max	mean	variance	standarddeviation	standarderror	skewness	kurtosis
11.52	12.57	12.0113	0.052421...	0.228957474769052	0.0228957474...	0.272723...	-0.55175...
11.74	12.63	12.0818	0.034988...	0.18705252327798...	0.0187052523...	0.433745...	-0.04749...
11.35	12.38	11.8811	0.044727...	0.21148773157326...	0.0211487731...	-0.27823...	-0.11634...
11.56	12.53	12.0774	0.037821...	0.194477388826233...	0.0194477388...	-0.27112...	-0.15905...

- 对四台包装机（前4列数据）做单样本t检验：
- 单样本t检验-字段选择为第1列数据-参数设置为备择假设greater、置信度0.95假设均值12
- 单样本t检验-字段选择为第2列数据-参数设置为备择假设greater、置信度0.95假设均值12
- 单样本t检验-字段选择为第3列数据-参数设置为备择假设less、置信度0.95假设均值12
- 单样本t检验-字段选择为第4列数据-参数设置为备择假设greater、置信度0.95假设均值12

阿里云

项目实施

假设检验



- 机器1的t检验，p=0.31，接受原假设，说明包装机有问题；
- 机器2的t检验，p值极小，拒绝原假设，说明包装机没有问题；
- 机器3的t检验，t值为-5.62，而临界值a=0.05对应的t值为-1.96，所以样本落入拒绝域，拒绝原假设，说明包装机有问题；
- 机器4的t检验，p值极小，拒绝原假设，说明包装机没有问题。

项目实施

- 回归分析

- 特征列为速度，标签列为包装大小，生成模型评估表和回归系数评估。
- 预测结果误差在[-0.5,0.5]范围。

项目总结

12、监控项目实施 → 13、项目总结

- 项目执行完成后，进行项目的经验、教训总结。

- 数据项目要根据**“项目的目标”**来评估绩效。
- 例如：
  - 一个数据项目的目标是提高数据查询的速度提高到每秒处理TB级查询。那项目的成功就必需按是否达到目标来衡量。
  - 后信用卡发行部门需要知道哪类客户最可能申请公司的信用卡，那用分析结果执行营销活动获取的客户数量是否有提高就是衡量的指标。而客户是否使用信用卡就不是衡量的指标。

项目总结表

Project Overviews

一、项目基本情况 Project Basic Info					
项目名称 project name:	基于工业数据的检测分析		项目编号 project code:	T0001	
制作人 prepared by:	王红		审核人 reviewed by:	张明	
项目经理 project manager:	张明		制作日期 data:	2017-9-10	
二、项目完成情况总结 Project fulfillment sum-up					
1、时间总结 1、time aspect					
开始时间: Start date:	2017-8-1	计划完成日期: Expected finish date:	2017-9-3	实际完成日期: Actual finish date :	2017-9-5
时间（差异）分析 time （discrepancy） analysis:					
多次选择算法，工具学习时间比较长；					
2、成本总结 2、cost （ discrepancy）analysis:					
计划费用: Expected expense:	￥1,0000 元		实际费用: Actual expense:	￥1,0000 元	
成本（差异）分析 cost（discrepancy） analysis:					
无差异					
3、交付结果总结 3、deliverables aspect:					
计划交付结果 expect deliverables:					
在 2017 年 8 月 20 日向客户汇报项目完成情况，让打消客户对公司供货能力的怀疑，增强客户对我司研发能力、工程管理能力的认可。					
实际交付结果 actual deliverables:					
客户于 2017 年 8 月 18 日收到我们项目的进展程度，已消除了疑虑，认可我司的供货、研发和工程管理能力，支持我司后续项目实施。					
未交付结果 undeliverables:					
无					
交付结果（差异）分析 deliverables （discrepancy） analysis:					
不涉及					
三、项目经验、教训总结 Ⅲ 、Project experience/lessons sum-up					
经验： 1、项目分工明确，责任清晰，进度设置合理； 2、项目组内沟通渠道顺畅，公司高层领导大力支持； 3、风险响应计划较为得当，避免了大的失误； 教训：选择熟悉的算法及工具，项目体系设计时就就要确定算法，从而避免在后期发生临时变更。今后的项目前期需求分析应该更准确。					
签字 signature 张明			日期 date 2017-9-10		

项目后分析

- 项目后分析评价应分析项目所达到和实现的实际结果，根据项目运营和未来发展以及可能实现的效益、作用和影响，评价项目的成果和作用。

项目目标达成状况。

ROI状况分析。项目的投入与产出的财务回报状况。

项目提高的空间。

分析方法贡献的绩效。

课程目录

1. 项目管理方法论
2. 数据项目实施流程
3. **数据项目设计方法**

数据项目设计

- 处理数据分析项目时，可从四个方面入手了解项目，进行数据项目设计。

项目目标

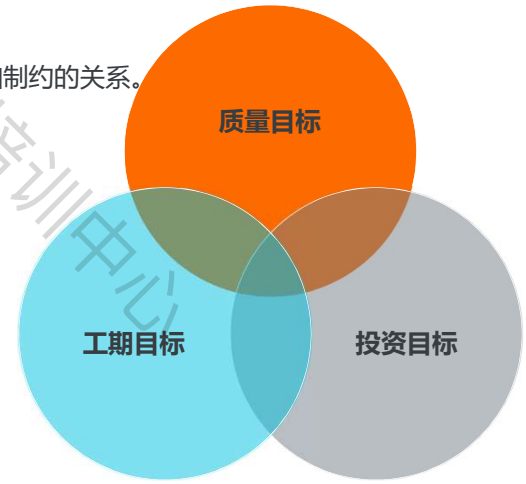
背景调查

分析范围

分析结果交互形式

项目目标

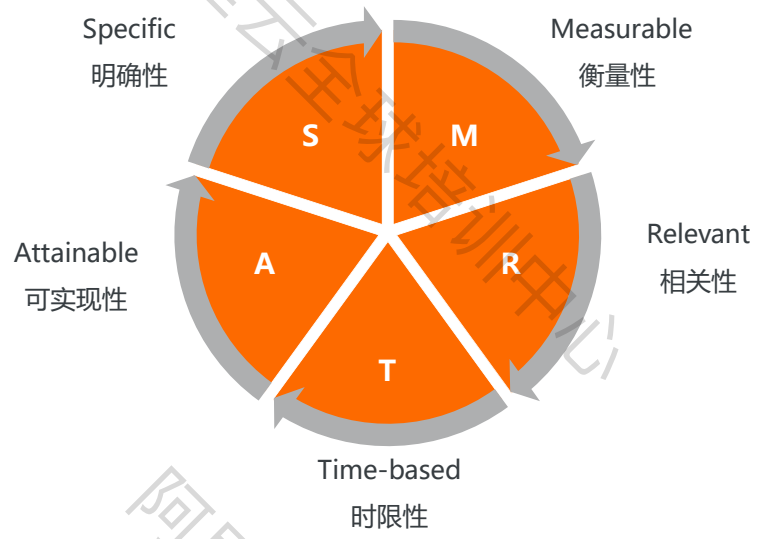
- 项目目标是指一个项目为了达到预期成果必须完成的各项指标标准。
- 三大的目标值由合同界定，彼此之间存在着相互联系和制约的关系。



目标确定过程

项目情况分析 → 项目问题界定 → 确定项目目标因素 → 建立项目目标体系 → 各目标的关系确认

项目目标原则



阿里云

目标制定方法

工作分解结构（WBS）

责任矩阵（responsibility matrix）

鱼骨图

阿里云

项目目标案例

- 以营销的数据分析项目为例，常见的项目目标有以下一些例子：
  - 市场机会分析，如市场占有率，成长机会，潜在客户等。
  - 目标市场选择，如市场细分、市场定位等。
  - 产品定位，如产品决策、定价、销售渠道决策等。
  - 促销优化。
  - 分析客户的购买习惯以发现市场机会。
  - 目标市场细分。
  - 品牌营销，如品牌知名度和广告知名度、品牌忠诚度等。
  - 潜在销售量的估计。
  - 广告效果研究。
  - 对经销商的绩效分析。

背景调查

范围分析

- 一个数据项目从其一成立开始，其分析的范围就必须要明确。数据涉及的范围基本有以下三种：

- 分析范围的选取需要遵循奥康姆剃刀原则，即“如无必要，勿增实体”，即可以用少数数据来说明事物的时候，却用了许多的数据，那就是浪费。

分析结果和交付形式

- 大多数数据项目的交付形式是以下形式为主：

- 当然，也有一些数据，可以通过口头、邮件等形式交付，通常是较简单的项目可以通过该形式交付。