Técnicas e Programação de Analytics

Ementa

Contextualizando Analytics ▫ Habilidade Hacker: ôxe! Não se assuste com isso! ▫ Conjunto de Dados (dataset) ▫ Linguagens de Programação para Analytics ▫ Tipos de Dados ▫ Estrutura Lógica Básica ▫ Hands On R ▫ Hands On Python

Conteúdo Programático

Parte 01

Parte 01

1. Contextualizando Analytics

- Data Science
- Data Analysis
- Predictive Analytics
- Análise orientada a Dados

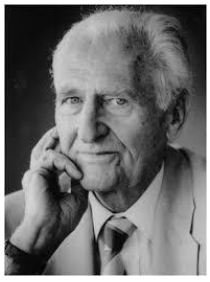
Contextualização

O Analytics é um campo abrangente e multidimensional que usa matemática, estatística, modelagem preditiva e técnicas de aprendizado de máquina para encontrar padrões e conhecimentos significativos nos dados registrados.

Hoje, adicionamos computadores poderosos à mistura para armazenar quantidades crescentes de dados e executar algoritmos de software sofisticados - produzindo os insights rápidos necessários para tomar decisões baseadas em fatos. Colocando a ciência dos números, dados e descoberta analítica para funcionar, podemos descobrir se o que pensamos ou acreditamos é realmente verdade, e produzir respostas para perguntas que nunca pensamos em perguntar. Esse é o poder da análise.

Por que a análise é importante?

Desde o primeiro projeto de coleta de dados populacionais conhecidos pelo governo sueco em 1749, até o registro de Florence Nightingale e análise dos dados de mortalidade na década de 1850, ao estudo do epidemiologista britânico Richard Doll sobre o câncer de pulmão e câncer na década de 1950, a análise de dados já era realizada há mais de 100 anos.



Se observarmos bem, cada um dos cenários acima exigiu uma resposta a uma pergunta até então não possível de ser respondida. Em 1700, os suecos queriam saber a distribuição geográfica de sua população para aprender a melhor maneira de sustentar uma força militar apropriada. Nightingale queria saber o papel que a higiene e o cuidado de enfermagem desempenhavam nas taxas de mortalidade. Doll queria saber se as pessoas que fumavam eram mais propensas a sofrer de câncer de pulmão.

Cada um desses pioneiros sabia que o instinto não era bom o suficiente. A análise de dados pode revelar correlações e padrões e com ela há menos necessidade de confiar em suposições ou intuição. E isso pode ajudar a responder os seguintes tipos de perguntas:

- O que aconteceu?
- Como ou por que isso aconteceu?
- O que está acontecendo agora?

- O que é provável que aconteça depois?

Com computadores mais rápidos e mais poderosos, a oportunidade é abundante para o uso de análises e big data. Seja determinando o risco de crédito, desenvolvendo novos medicamentos, encontrando maneiras mais eficientes de fornecer produtos e serviços, prevenindo fraudes, descobrindo ameaças cibernéticas ou retendo os clientes mais valiosos, a análise pode ajudá-lo a entender sua organização - e o mundo ao seu redor.

Assim, Analytics é descobrir informações significantes a partir dos dados disponíveis e alguns especialistas costumas dividir em 4 sub categorias:

Descritiva: diz o que está acontecendo.

Diagnóstica: diz porque está acontecendo

Preditiva: diz o que irá acontecer no futuro.

Prescriptiva: diz o que fazer para mudar o que está acontecendo.

1.1 Data Science

A ciência de dados é uma área multidisciplinar que utiliza métodos científicos, processos, algoritmos e sistemas para extrair conhecimento e insights a partir de dados estruturados e não estruturados.

É o campo de estudo que combina expertise e domínio sobre um dado assunto, habilidades de programação e conhecimentos sobre matemática e estatística para extrair insights a partir dos dados. Os cientistas de dados normalmente aplicam algoritmos de aprendizagem de máquina a números, textos, imagens, vídeos, áudio e etc, para produzirem sistemas baseados em inteligência artificial para executares tarefas que requerem inteligência humana. Estes sistemas geram insights para que analistas usuários do negócio transformem os mesmos em valores tangíveis.

1.2 Data Analysis

É a descoberta, interpretação e comunicação de padrões nos dados e o processo de aplicar estes padrões em direção à tomada efetiva de decisões. Em outras palavras, analytics pode ser entendido como o tecido conectivo entre dados e tomadas de decisões dentro de uma organização.

É o resultado das análises sistemáticas de dados ou de estatísticas.

1.3 Análise Orientada a Dados (Data Analytics)

É o processo de examinar conjunto de dados de forma que seja possível tirar conclusões sobre as informações que eles contém. Atualmente isto é feito com

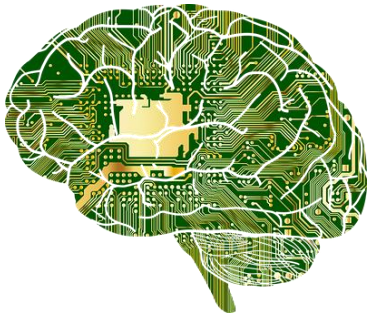
o uso de algoritmos, softwares especializados e hardwares com bom desempenho computacional.

As técnicas e tecnologias para a Análise Orientada a Dados são amplamente utilizadas para fins comerciais e industriais visando possibilitar às organizações uma tomada de decisão mais assertiva para os negócios e também por pesquisadores e cientistas, para poder verificar ou comprovar modelos, teorias e hipóteses.

Como termo, Data Analytics refere-se predominantemente como um leque de ferramentas, desde o básico do BI, como relatórios e processamento analítico (OLAP) até as formas mais avançadas de análise. Desta forma, isto é similar ao Business Analytics, outro termo guarda-chuva para análise de dados, com a diferença que este último é orientado aos usuários do nível de negócio, enquanto Data Analytics possui um foco mais abrangente. A visão expansiva do termo não é universal, porém, em alguns casos, pessoas usam Data Analytics para designar uma análise mais avançada, tratando BI como uma categoria separada.

2. Habilidade Hacker: ôxe! Não se assuste com isso!

- A melhor definição
- Desenvolver para crescer
- Visão Exploratória: Data Discovery



Na língua inglesa, a palavra deriva do verbo to hack, que significa "cortar grosseiramente", por exemplo com um machado ou facão. Usado como substantivo, hack significa "gambiarra" — uma solução improvisada, mais ou menos original ou engenhosa.

Esse termo foi apropriado pelos modelistas de trens do Tech Model Railroad Club na década de 1950 para descrever as modificações que faziam nos relês eletrônicos de controle dos trens. Na década de 1960, esse termo passou a ser usado por programadores para indicar truques mais ou menos engenhosos de programação, por exemplo usando recursos obscuros do computador. Também foi usado por volta dessa época para manipulações dos aparelhos telefônicos com a finalidade de se fazer chamadas grátis.

2.1 A melhor definição

Aquele(a) que se esforça em encontrar meios para solucionar, melhorar ou extrapolar algo sem se limitar à sua caixa de ferramentas.

2.2 Desenvolver para crescer

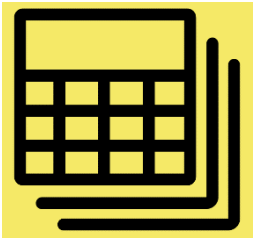
Pratique mais e mais vezes para desenvolver suas habilidades. Não se limite ao que é puramente técnico e puramente teórico. Busque identificar onde está o seu limite. Você não tem obrigação de conhecer tudo, mas deve estar disponível para lidar com o desconhecido. As habilidades para trabalhar com análise de dados

são desenvolvidas ao estar com a mentalidade correta para lidar com a imperfeição ao redor do mundo que produz ou disponibiliza os dados. Busque fontes de dados abertas disponíveis na Internet para poder praticar várias formas de análise, ter contato com algoritmos e ferramentas. Desenvolva a sua curiosidade para tratar dados.

2.3 Visão exploratória

Atue como uma sonda, busque encontrar o que não é facilmente percebido. Hoje é senso comum que precisamos pensar fora da caixa. Utilize técnicas de exploração de dados, como busca de padrões, correlações, agregações. Se for necessário utilize gráficos, editores de texto, dicionários de dados, dicionários linguísticos, postulados, sites de referência etc. Extraia as características que você considerar importante para construir seu modelo de análise, pois alguém irá te perguntar sobre suas decisões. Dados disponíveis em grande volume precisam ser avaliados de forma amostral, não avance no todo antes de avaliar o perímetro daqueles dados. Lembre-se: a exploração precisa começar de algum ponto e não pelo todo.

3. Conjunto de Dados (dataset)



- Fontes Internas
- Fontes Externas
- Cuidado com a Correnteza

Dataset é uma respresentação de um conjuto de dados qualquer, podendo ser estruturado, semi-estruturado ou não estruturado. Para quem trabalha com análise de dados, os datasets representam a matéria-prima a ser utilizada de forma exploratória.

Quanto ao tamanho, os datasets podem variar de acordo com volume.

Os datasets são normalmente originados de fontes internas ou externas.

3.1 Datasets de Fontes Internas

Os datasets de fontes internas são aqueles onde os dados são originados de sistemas em uso na organização.

Exemplos de fontes internas: sistemas de folhas de pagamento, sistemas contábeis, datawarehouse, sistemas financeiros, sites corporativos e portais da organização etc.

3.2 Datasets de Fontes Externas

Os datasets de fontes externas são normalmente originados de bancos de dados de sistemas de parceiros, fontes abertas, data selling, operações de data

exchanging (como operações bancárias, emissões de notas fiscais eletrônicas, entre outros), mídias sociais, etc.

Exemplos de fontes externas: extração de tuites, consumo da API do facebook, consumo da API da secretaria da fazenda, acesso a log do servidor de hospedagem de um provedor terceirizado, consumir um serviço na internet, etc.

3.3 Cuidado com a Correnteza

Atualmente temos uma enchorrada de dados circulando ao nosso redor e através de diversos serviços de dados numa correnteza. O termo mais comum para isso é Streaming de Dados. Muitos serviços de vídeo sob demanda como o Netflix utilizam este tipo de distribuição de dados para seus usuários. Entretanto, realizar o Data Streaming não é uma coisa trivial, pois uma de suas características é o uso de múltiplas fontes de origem de dados para serem canalizados num único canal e de forma lógica, formando ou não as famosas "bolhas". As bolhas são universos individuais para algum destinatário dos dados. Por exemplo, quando você faz uma busca na Internet utilizando os mesmos termos que um amigo seu que fez a busca no mesmo momento que você, resultados diferentes poderão aparecer a depender do que o seu universo tenha como característica de interesse de acordo com o seu histórico avaliado pelo buscador. Por trás dos Data Streaming existe muita inteligência de estruturação para os dados canalizados, fazendo com que esse seja um dos ambientes mais ariscos para os analistas de dados. Assim como nos rios e oceanos, onde a correnteza pode mudar a qualquer momento, o mesmo pode acontecer com os dados originados nestes serviços, podendo impactar quaisquer tipos de análises.

4. Linguagens de Programação para Analytics

- Vixe! Eu nunca programei, e agora?
- R
- Python
- Outras
- Limitações das linguagens

Além de ser possível utilizar ferramentas comerciais prontas para usar, o famoso plug-and-use, muitas vezes estas mesmas ferramentas "deixam a desejar" quando algum tipo de análise foge do modelo empacotado. Quase que por via de regra, estas ferramentas oferecem conectores para o uso com algum tipo de

linguagem do mundo Analytics como: R, Python, Scala, Java, Ruby, etc. Por este motivo, uma das habilidades hacker que o analista de dados deve ter é o conhecimento básico em qualquer uma das linguagens que oferecem poderes mágicos para a análise de dados.

4.1 Vixe! Eu nunca programei, e agora?

A prática será sempre a sua melhor aliada caso você nunca tenha feito nenhum programa de computador. Hoje temos um grande leque de opções para aprender qualquer coisa e a Internet é a principal fonte de conhecimento que nós temos. Muita gente que quer trabalhar com analytics entende de programação e não entende de negócio ou não entende de processos. A multidisciplinaridade é uma característica importante para quem atua analisando dados e para que possa ter os insights necessários sobre os mesmos, adquirir conhecimentos sobre programação virou coisa básica para profissionais de diversas áreas que possuem interesse ou necessidade de realizar análise de dados.

4.2 A Linguagem R

R (erre) é o nome de uma linguagem estatística e também de um ambiente de estatística computacional. É muito utilizada por estatísticos, analistas de dados e por mineradores de dados. Possui uma vasta biblioteca de pacotes que expandem a área de aplicação da linguagem. Foi lançada como software livre em 1995 pelos seus criadores Ross Ihaka e Robert Gentleman que trabalhavam juntos e eram colegas no departamento de estatística na Universidade de Auckland. Os dois compartilhavam o interesse por estatística computacional e viram a necessidade de um melhor ambiente de software para o laboratório da área. Os dois não acharam nenhum software compatível no mercado e começaram a tentar desenvolver um eles mesmos. (Wikipedia)

Segundo o próprio site da linguagem, ela é similar à linguagem S e pode ser considerada com uma implementação diferente, porém muitos códigos escritos em S podem ser executados em R sem nenhuma alteração e descreve como suas principais características do seu ambiente de trabalho, o seguinte:

- Recursos efetivos para manipulação e armazenamento de dados;
- Uma suíte de operadores para realização de cálculos em arrays e matrizes;
- Uma ampla, coerente e integrada solução de ferramentas intermediárias para análise de dados;
- Recursos gráficos para exibir e analisar dados tanto na tela como, em papel;
- Uma linguagem interpretada bem desenvolvida, simples e efetiva, que inclui condicionais, loops, funções recursivas definidas pelo usuário e recursos de entrada e saída de dados;
- Está disponível para ambiente Windows, Linux e MacOS.

A linguagem R possui licença livre e já é adotada como complemento de produtos de grandes players de mercado, como a Microsoft e a Oracle.

Seu uso na área de Data Analytics se popularizou com a adoção da mesma nos grandes centros acadêmicos, inicialmente para o estudo e prática da estatística e depois pela grande variedade de pacotes que surgiram e simplificaram o trabalho de análise.

4.3 A Linguagem Python

É uma linguagem de multipropósito, ou seja, pode ser utilizada para fazer uma quantidade grande de coisas, desde jogos, sistemas, sites e até sistemas de aprendizagem de máquina. Foi criada por Guido van Rossum e lançada em 1991. A linguagem dispõe de um interpretador de comandos e uma vasta lista de pacotes específicos para cada tipo de propósito. (Wikipedia)

No site oficial, as seguintes características são colocadas em destaque:

- É para o desenvolvimento de aplicações para a Web e Internet;
- É científica e numérica é para a análise de dados;
- Voltada para o ensino de programação de computadores;
- É para o desenvolvimento de aplicações Desktop;
- É para apoiar o desenvolvimento de softwares;
- É para o desenvolvimento de aplicações de negócio;
- É para Windows, Max OS e a família Unix.

É fortemente utilizada por grandes players de mercado como Google, Microsoft, Oracle, Facebook e Twitter para finalidades diversas. Pode ser executada sobre outras plataformas como Java e .NET.

Sua popularização passou a crescer muito após a Google adotá-la como principal linguagem para desenvolvimento da sua pilha de serviços de Inteligência Artificial e liberando suas bibliotecas para uso livre da comunidade.

Python hoje possui uma lista bem variada de bibliotecas/pacotes voltados para a análise de dados.

4.4 Outras linguagens

Linguagens como Java, Ruby, Scala também são muito utilizadas para análise dados e possuem também uma variada lista de pacotes para analytics. Entretanto, ainda carecem de uma comunidade de entusiastas e fãs que as coloquem no mesmo nível de divulgação e uso que R e Python possuem, mas isto não quer dizer que elas não são úteis para quem trabalha nessa área.

4.5 Limitações das Linguagens

Você utilizaria um Fusca para fazer uma mudança? Alguns irão dizer que sim, outras até confirmarão que já fizeram e outros dirão que não, pois não é o mais indicado. Mas peraí! O Fusca nasceu para ir à guerra! Você já pensou em colocar um guincho no Fusca e colocar os móveis num trailer para ele rebocar?!?!?

Isto é só uma brincadeira, mas quando falamos em linguagens o mesmo acontece e por vários motivos. Entretanto, as maiores limitações que podemos encontrar nas linguagens usadas no mundo analytics estão relacionadas ao tipo de aplicação final do software executável que será colocado em produção.

Muito codificadores subestimam as limitações técnicas de bibliotecas e pacotes disponíveis para as linguagens e só descobrem o problema após o modelo ser colocado em produção. Busque inicialmente conhecer os diversos tipos de limitações das linguagens e suas bibliotecas/pacotes e confronte-as com os requisitos de volumetria de dados, escalabilidade e arquitetura do ambiente do software, pois isto te ajudará a repensar ou readequar a sua estratégia de criação do modelo analítico e consequentemente do seu software. Você sempre encontrará pessoas que preferem um tipo de linguagem em detrimento de outra, mas busque seguir o caminho da racionalidade para encontrar a melhor solução. Muitas vezes as limitações de uma biblioteca são confundidas como limitação de uma linguagem e cabe ao analista fazer este tipo de avaliação. As linguagens estão sempre em evolução mas também é preciso não ter pressa em sair atualizando o seu ambiente de desenvolvimento para a última versão, pois muitos pacotes e bibliotecas poderão não funcionar com ela. Espere sempre a homologação dos desenvolvedores que as criaram e tenha cautela após a homologação, pois bugs podem surgir assim como pode ocorrer a mudança da sintaxe de algumas funções que você pode ter usado em sua análise.

5. Tipos de Dados

- Estruturados
- Semiestruturados
- Não estruturados
- Tipagem de Dados
- Data Wrangling/Munging/Preparation

5.1 Dados Estruturados

São dados que estão contidos dentro de uma estrutura única e fixa de forma matricial, através de linhas, colunas e os dados são submetidos a regras de tipagem.

Exemplo: Tabelas de Bancos de Dados, Arquivos CSV Single, planilhas eletrônicas.

5.2 Dados Semi-estruturados

São dados que estão contidos dentro de estruturas hierarquizadas, onde múltiplas estruturas são contidas dentro de um único arquivo, podendo existir a presença de dados não estruturados. Os dados semiestruturados são muitas

vezes considerados estruturados quando não incorporam campos opcionais que contenham dados desestruturados.

Exemplo: Arquivos CSV Master/Detail e Arquivos XML e planilhas eletrônicas.

5.3 Dados não estruturados

São dados que possuem uma organização não clara ou onde o esforço para se determinar uma organização para os mesmos possui um custo operacional muito elevado. Muitas vezes estão contidos dentro de campos de dados estruturados.

Exemplo: textos, vídeos, imagens

5.4 Tipagem de Dados

A tipagem de dados pode ser categorizada quanto ao domínio do dado e quanto à natureza das modificações que serão necessárias para que o mesmo se torne analisável.

Os domínios mais conhecidos e utilizados são:

Numéricos: composto de números fracionários ou inteiros que podem ser utilizados para fins de cálculos e filtros;

Data: composto de datas válidas que podem ser utilizados para fins de cálculos e filtros;

Lógicos: normalmente representa uma faixa de valores de um domínio boleano como: Verdadeiro e Falso, Sim ou Não, 1 ou 0, etc.

Caractere: o domínio deste tipo de dados é formado pela tabela de caracteres adotada pelo formato da fonte de dados e normalmente representa um caractere;

String: formado por um valor que pode ser formado por uma determinada quantidade de caracteres que fazem parte da tabela de caracteres adotada pelo formato da fonte de dados;

Blob: possui tamanho variável e pode possuir qualquer caractere que faz parte da tabela de caracteres adotada pelo formato da fonte de dados;

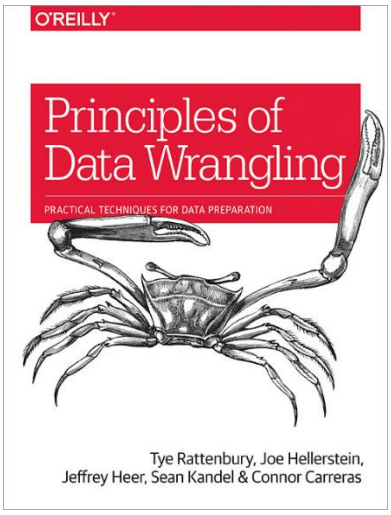
Quanto à natureza das modificações a serem realizadas, os dados podem ser:

Raw (Crú ou Bruto): são dados que ainda não possuem um esquema ou uma estrutura aplicada e futuramente precisará ser preparado para uso.

Processed (Processado): é o estado posterior a uma preparação feita sobre o dado Raw, após o mesmo ter sido estruturado num determinado esquema.

Cooked (Cozido): são dados que possuem algum tipo de análise estatística, filtragem, sumarização ou agregação realizada sobre ele para ser consumida em algum processamento, tratamento ou análise futura.

5.5 Data Wrangling/Munging/Preparation



Antes de qualquer coisa, precisamos avaliar o que estes termos têm a ver com tipos de dados. Gostaria de resumi-los na seguinte forma: mastigar dados. Vamos ver por que isso: uma das coisas consideradas mais chatas para se fazer em relação a análise de dados é a preparação dos mesmos para que possam ser analisados. Temos sempre que levar em consideração que as fontes são diversas e nem sempre os dados disponíveis possuem qualidade. Algumas vezes os dados estão em fontes que deixam os analistas boquiabertos, como por exemplo, uma tabela que está disponível dentro de um arquivo PDF e ocupa cerca de 15 páginas. Você pode estar se perguntando: por que não disponibilizam estes dados diretamente da fonte que gerou o arquivo? Esta é uma boa pergunta, mas acredite em mim. No mundo dos negócios isso nem sempre é possível e sempre olharão para você como alguém que possui habilidades excepcionais do tipo: usa o copiar e colar do Excel e remonta a tabela. Juro que já fiz isso!

Mas vamos voltar aos significados via tradução:

Data Wrangling: Disputa por Dados = Você versus a Pseudo Fonte versus Área de Negócio

Data Munging: Dados Desagradáveis de se obter

Data Preparation: Preparação de Dados

Aqui é importante deixar explícito o seguinte: isto não é o processo de qualidade de dados! É apenas o procedimento de obtenção de dados. E o que isso tem a ver com Tipos de Dados? Os dados obtidos deste procedimento serão disponibilizados para a construção de alguma análise e isto requer uma visão sobre o nível de estruturação dos mesmos e como deverão ser devidamente tipados para esta futura análise. Qualquer analista de dados experiente sabe que todo o procedimento de Data Wrangling representa uma quebra de planejamento nas questões relacionadas a estimativas de prazos, pois o nível de incerteza é bastante alto quando estamos lidando com uma pseudo fonte de dados e, nem sempre a última solução poderá ser utilizada plenamente para resolver as questões de uma nova demanda.

Muitas vezes, as ferramentas utilizadas no processo de limpeza dados também são utilizados para a estruturação e enriquecimento de dados brutos em um formato desejado para melhor tomada de decisões em menos tempo. O Data Wrangling está cada vez mais onipresente nas principais empresas que trabalham com dados. Os dados se tornaram mais diversificados e não estruturados, exigindo maior tempo gasto na coleta, limpeza e organização deles antes de uma análise mais ampla. Ao mesmo tempo, com dados apoiando

praticamente todas as decisões de negócios, os usuários corporativos não estão dispostos a esperar por recursos técnicos obterem os dados preparados. Após longos anos de discussão se isso deveria ser papel dos Data Analysts, um novo entendimento surgiu, colocando cada vez mais a responsabilidade da realização do Data Wrangling nas mãos daqueles que mais conhecem os dados, pessoas que estão no nível do negócio. Com isso, várias ferramentas foram surgindo e promovendo um modelo de autoatendimento ou self-service, visando um afastamento da preparação de dados liderada pela TI para um modelo mais democratizado de Data Wrangling. Esse modelo de autoatendimento permite que os analistas lidem com dados mais complexos com mais rapidez, produzam resultados mais precisos e tomem decisões melhores na execução dos projetos.

Algumas ferramentas para Data Wrangling:

Pandas (pacote Python), DLP-R e TIDY-R (pacotes R), Mr. Data Converter, CSVKit, DataWrangler, Tabula e OpenRefine.

6. Estrutura Lógica Básica

- Conhecer o problema ou Objetivo de Negócio
- Os dados existem?
- Conhecer o propósito dos Algoritmos
- Você falou em aprendizagem de máquina?
- Construa o modelo: treino é treino e jogo é jogo
- Não invente a roda: biblioteca é o lugar!
- Produza sua análise
- Valide sua análise
- Apresente sua descoberta

Com o passar dos anos a área de data analytics vem adquirindo novas perspectivas em direção ao amadurecimento dos processos práticos, possibilitando o surgimento das melhores práticas em uso.

As estruturas lógicas básicas que atualmente norteiam o trabalho dos analistas de dados normalmente possuem duas perspectivas: end-to-end (fim-a-fim) e a perspectiva do analista codificador.

No diagrama a seguir, temos o processo representado pela perspectiva end-to-end que é voltada diretamente para o cerne do negócio e possibilita a descrição macro do trabalho a ser executado em suas diferentes fases.

No próximo diagrama, temos a representação do processo sob a ótica do analista que irá codificar o algoritmo para a análise.

Observe que ambos são parecidos, porém o segundo fornece um nível de detalhamento maior para quem codifica. Estas visões são importantes para os diversos níveis de interessados nos insights da análise.

6.1 Conhecer o problema ou Objetivo de Negócio

Um analista de dados deve desenvolver a habilidade de entender como funciona o negócio ou o objetivo de negócio que gera a demanda de análise. Pois desta forma, será capaz de falar a mesma linguagem dos analistas de negócio e sugerir meios viáveis para a obtenção de insights. Neste ponto, saber fazer a pergunta certa irá poupar horas de trabalho de todos os envolvidos.

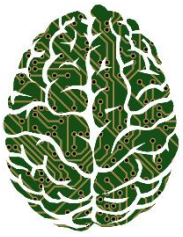
6.2 Os dados existem?

Esta pergunta vale um milhão (US$). Não se assuste se não houver resposta para esta pergunta. Muitas vezes a resposta vem incompleta e você precisará lidar com isso, atuando como um detetive, procurando investigar quais as prováveis fontes de dados que poderiam ou deveriam existir para que uma análise seja devidamente preparada. Lidar com este tipo de situação é muito comum e significa que você precisará desenvolver outra habilidade: a paciência.

6.3 Conhecer o propósito dos Algoritmos

Hoje temos disponível para uso uma grande quantidade de algoritmos pré-configurados para utilizarmos com nossos dados e, independente da linguagem ou software que estivermos utilizando, precisamos saber para que serve cada um deles. A linguagem e o software são meras ferramentas de trabalho e poderemos utilizar o que estiver disponível para uso. Entretanto saber qual o melhor algoritmo a ser utilizado para se alcançar um objetivo de análise é o maior diferencial de um analista de dados e isto requer muito estudo teórico e prático sobre a lógica por trás de cada um deles. Hoje falamos muito em aprendizagem de máquina, modelos supervisionados e não supervisionados e é papel do analista fazer a melhor escolha para a construção do modelo analítico a ser implementado.

6.4 Você falou em aprendizagem de máquina?



Sim! É bom saber quando devemos utilizá-la. Sempre que houver a necessidade de automação da construção de um modelo analítico onde haja a necessidade de tomar decisões com base nos padrões aprendidos com os dados e com o mínimo de interferência humana; use-a! Mas cuidado! A obtenção de insights neste caso requer um comportamento interativo entre o usuário e o algoritmo, típico de uma ação de pergunta e resposta.

6.5 Construa o modelo: treino é treino e jogo é jogo?

Cuidado com o senso comum que costuma dizer que treino é treino e jogo é jogo, pois isto pode te levar a um equívoco que pode ser irreparável. O principal motivo para se treinar um algoritmo é o aperfeiçoamento das nossas análises. Um algoritmo que não foi devidamente treinado para atingir um objetivo está fadado a produzir distorções da realidade, levando a tomadas de decisões que podem custar a vida de uma organização e até mesmo de pessoas. Pense como um piloto de avião que inicia sua vida em simuladores e começa a adquirir horas de vôo praticando em aviões menores de forma assistida por um tutor e ao pilotar de verdade, seus conhecimentos são colocados à prova desde o momento que entra na aeronave, pois a decisão de decolar, que é uma das tarefas mais arriscadas (assim como a aterrisagem) pode por em risco a vida do piloto e a de seu instrutor. Portanto, devemos aprender que a preparação dos dados é diferente do treinamento do modelo a ser usado pelo algoritmo. O treinamento deve ser feito com dados representativos da realidade, seja de forma amostral ou total, para que posteriormente possam ser validados. Sua mentalidade deve ser sempre: treino é jogo e jogo é treino!

6.6 Não invente a roda: biblioteca é o lugar!

Muitos analistas de dados consomem muito tempo codificando “do zero” algoritmos de aprendizagem de máquina, algo que não se justifica quando a maioria das ferramentas disponíveis (linguagens e softwares especializados) já dispõem de várias bibliotecas com pacotes de algoritmos que são constantemente atualizados com otimizações para fins de performance e economia de recursos computacionais, além de oferecer formas de escalabilidade no processamento dos dados.

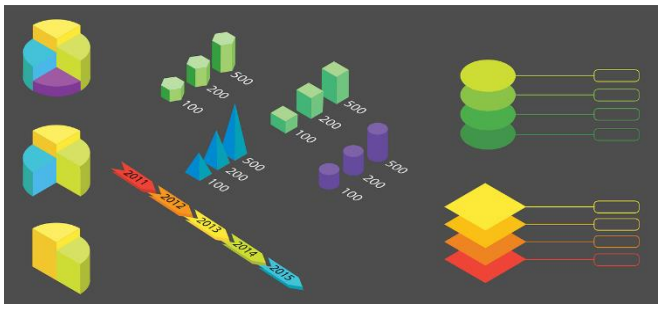
6.7 Produza sua análise

Não tenha medo de estudar os dados, mas busque se proteger, garantindo que os dados disponíveis estão de acordo com as normas ou regras de *compliance* disponíveis. Cabe aos analistas de dados olhar para as fontes de dados e iniciar uma longa rodada de sabatina com o conteúdo disponível. Faça de suas perguntas as suas aliadas ao imaginar cenários mesmo que surreais, mas que te levem a construir uma hipótese a ser testada. Estruture o que for preciso, faça a preparação dos dados e reserve uma quantidade deles para a fase de testes (normalmente 30% de um dataset).

6.8 Valide sua análise

Seguir caminhos tendenciosos é um dos riscos a serem gerenciados durante o processo de análise de dados e é importante que você faça testes com dados separados para esta finalidade e também com uma nova amostragem (dados frescos) para garantir que sua análise está adequada e pronta para ser apresentada.

6.9 Apresente sua descoberta



Os insights de uma análise devem ser apresentados para os usuários do nível de negócio tão logo a fase de testes e validação tenha sido concluída. Mas tome cuidado: hoje o mundo é dos infográficos e a expectativa em torno dos analistas de dados é que a saída da análise seja mostrada em alto nível. Portanto, seja criativo ao apresentar os insights. As linguagens e ferramentas já dispõem de mecanismos de apresentação, mas muitas vezes são pouco atrativos para quem não codifica. Por isto, muitas vezes você terá que exportar os dados para outro ambiente que possibilite vender seu trabalho de forma que deixe seus demandantes com vontade de perguntar: você pode copiar isto para mim? Quero apresentar para a diretoria.

Obs: os mecanismos de visualização estão se aperfeiçoando cada vez mais e é questão de tempo para que isso se torne algo de alto nível, sem a necessidade de utilizar recursos secundários.