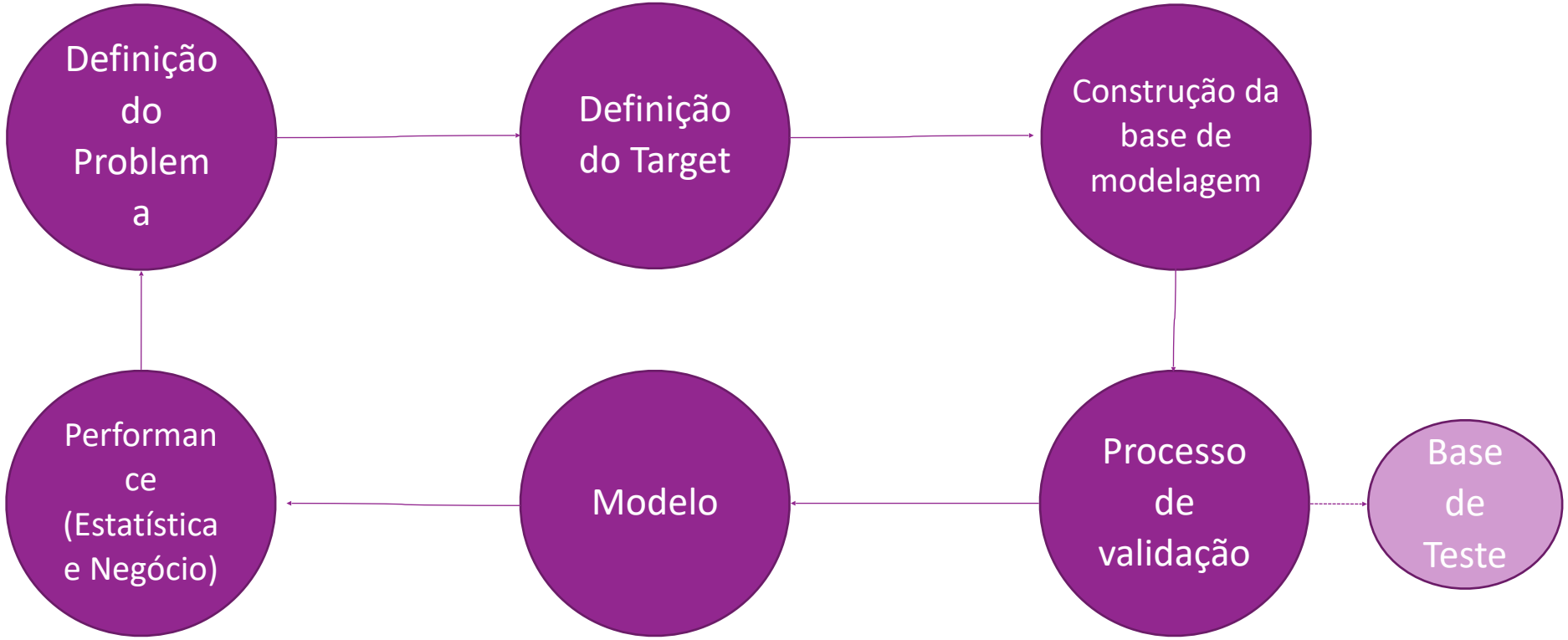
Meet Up – Machine Learning

03/12/2015

Etapas em um projeto de machine learning



Definição do Problema

Definição do Problema

Temos que definir exatamente o que queremos solucionar. Ex:

- Medir o risco de inadimplência de um cliente.
- Prever os gastos mensais para a definição de um limite ótimo.
- Informar ao atendente qual o possível problema que o cliente que está ligando irá falar a respeito.

Trabalho deve ser feito em conjunto com o analista que irá usar o modelo.

Definição do Target

Definição do Target

- Com o problema que queremos solucionar definido, temos a etapa de definição da variável resposta. Extremamente importante no processo!

- Ex.:

- O que representa o risco de inadimplência? Não pagamento parece ok, mas em quanto tempo? 10 dias de atraso é ruim? Quantos meses de observação vou olhar?

- O que uso para prever os gastos no próximo mês? O gasto do mês anterior? E efeitos sazonais?

Construção da base de modelagem

Construção da base de modelagem

- Usamos todos os dados (variáveis) disponíveis? Será que o signo da pessoa importa? Ou o nome?
- Todas as variáveis devem estar disponíveis no momento da tomada de decisão. Ex: um modelo que irá decidir a aprovação ou não de um cliente só poderá usar variáveis disponíveis naquele momento. Não importa se a renda atual é R$5.000,00 se nom momento da aprovação inicial a renda era R$2.000,00.
- Uma arquitetura / modelagem de banco de dados apropriada ajuda a garantir isso (histórico).

Processo de validação

Processo de Validação

Uma das etapas mais importantes. É graças a esta etapa que conseguimos testar o efeito do modelo na prática. Será que o nosso modelo terá o mesmo comportamento com novos dados?

Há diferentes estratégias de validação:

- Base de treino e teste.
- Base de treino, validação e teste.
- Base out-of-time. Útil para checar se o modelo será estável no tempo.
- Validação cruzada.

Processo de Validação

Estratégia usual é separação em três bases: treino, validação e teste.

- Treino: base utilizada para a construção do modelo.
- Validação: base separada que pode ser utilizada para otimizações do modelo.
- Teste: base que não é utilizada para nada. Quando iteramos o suficiente no modelo e achamos que o modelo está pronto para ser utilizado, medimos a performance nesta base. Tende a ser performance mais próxima da encontrada em produção.

Performance (estatística e negócio)

Performance (estatística e negócio)

- É sempre sugerido definir inicialmente qual métrica estatística será utilizada para verificar a performance do modelo.
- Da mesma forma as métricas de negócio. O modelo está conseguindo solucionar nosso problema?
- Trabalho em conjunto com o analista que irá desenvolver o modelo.
- Padronização para as próximas versões do modelo.

Modelo

Discutiremos adiante as etapas de construção do modelo.

Etapas do Modelo

Análise Descritiva das Variáveis

- Distribuição e correlação de cada variável com o target.
- Nesta etapa checamos a quantidade de valores missing e possíveis valores aberrantes.
- Obter *insights* para novas variáveis, bem como na melhor forma de tratar a variável com a resposta (é linear, quadrática?).
- Bugs no processo de construção dos dados também são encontrados aqui.

Missing e Outliers

- Como procedermos com valores missing (ex: nulo, strings vazias)? Algumas técnicas de machine learning são capazes de lidar com isso automaticamente, mas muitas não. Qual a melhor forma de fazer isso?
- Outliers podem impactar significativamente o modelo. Será que temos que fazer algo? É um risco em produção?

Criação de novas variáveis

- Com as análises anteriores ou conhecimento do negócio, podemos concluir que duas variáveis podem ser combinadas ou que uma transformação é mais apropriada.
- Muitas pessoas tendem a ignorar esta etapa, embora possa produzir grandes melhorias no modelo.

Feature Engineering

- Qual a melhor forma de tratar cada variável?
- Categorizar é uma boa alternativa?
- Devo usá-la de forma linear?
- Realizar uma transformação não-linear? Ex: spline.

Feature Selection

- Tenho centenas de variáveis. Devo usar todas? Quais são as mais importantes?
- Há algoritmos que se beneficiam de mais variáveis, outros algoritmos podem apresentar problemas (ex: instabilidade, multicolinearidade, tempo de execução, memória).
- Qual a melhor abordagem para escolher o melhor subconjunto de variáveis?

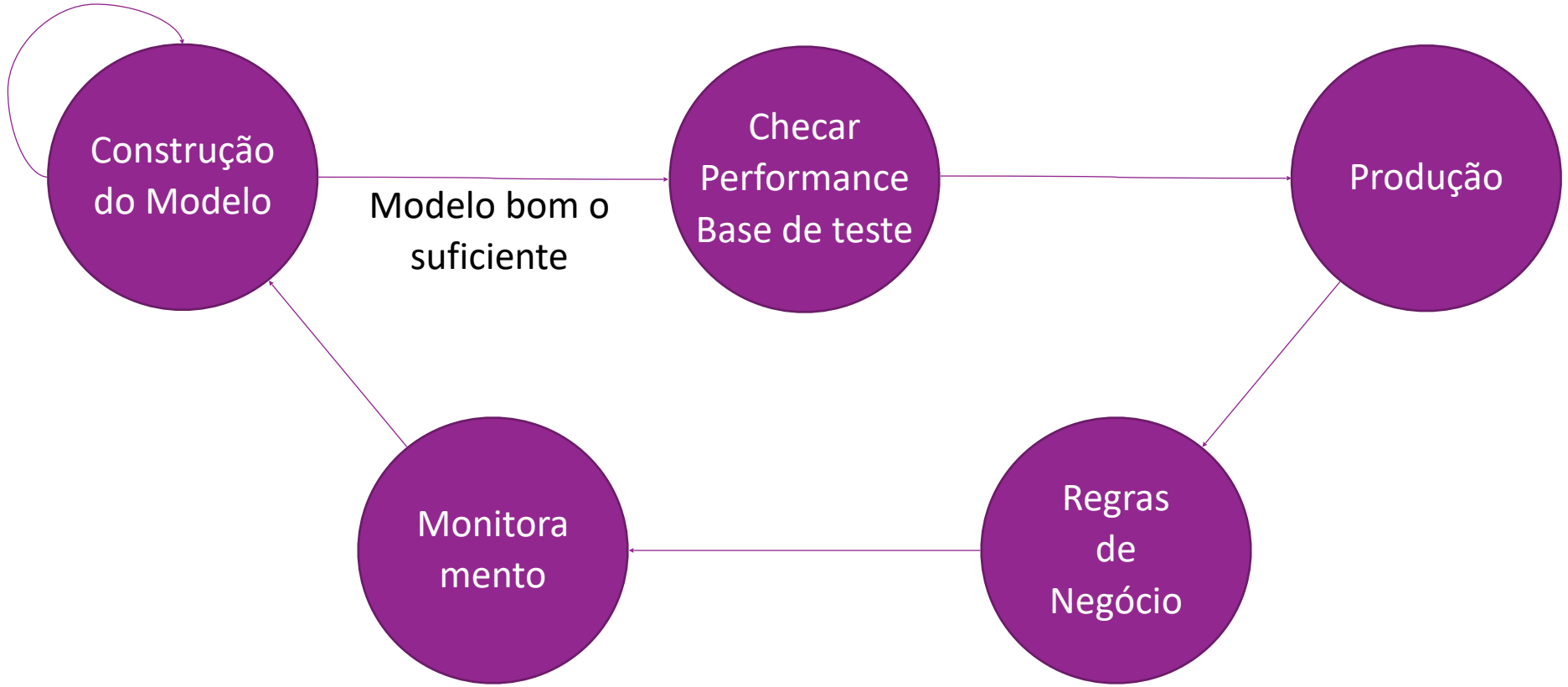
Modelo

- Qual técnica de modelagem (ex: Linear, Random Forest, Boosting, Redes Neurais)?
- Trade-off: preciso interpretar os meus resultados (ex: linear), ou posso usar um algoritmo mais poderoso (ex: rede neural)?
- Há alguma restrição regulatória? Ex.: O banco central exige alguma interpretação?

Implementação

Usando o modelo para decisões de negócio

Etapas na implementação



Base de Teste

Estamos contentes com o poder preditivo / estabilidade nas bases de treino e validação.

Próximo passo: checar estas métricas na base de teste.

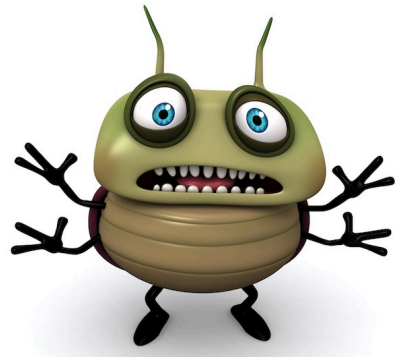
Base de Teste

Métricas

Performance similar aos obtidos durante o treino → Bom!

Performance muito inferior → Sinal de overfitting!

Performance muito superior → Sinal de bug!



Produção

Com um modelo estimado e aprovado, como usamos as predições em produção?

Produção - Alternativas

- Re-implementar o modelo em um sistema de produção (ex: modelo estimado usando R, sistema em Java)?
- Integrar a base de código do modelo em um sistema de produção (possível quando se usa a mesma linguagem/stack)?
- Exportar os parâmetros do modelo para um formato portável (PMML)?

Produção - Requisitos

- O modelo é uma função determinística (ex: geradores de números aleatórios são iniciados com uma semente que só depende das variáveis de entrada)?
- Execução em batch ou streaming? O modelo executa uma query (data pull) ou outro sistema fica responsável por preparar as variáveis (data push)?
- As variáveis estão no formato / tipo esperado?
- Como tratar valores ausentes? Valores aberrantes?
- Em caso de erro, consegue reproduzir as condições necessárias para recalcular a predição?
- Há tempo limite para resposta? Restrições no uso de CPU/memória?

Regras de Negócio

Temos um modelo prevendo uma determinada variável para cada observação (ex: probabilidade de inadimplência de um cliente, motivo de reclamação de um email).

Como usar esse dado para decisões de negócio?

Regras de Negócio

Score de crédito:

- Score > X = Aprovado
- Y < Score < X = Indecisos (ex: olhar mais dados, rodar um modelo mais poderoso)
- Score < Y = Rejeitado

Qual a perda esperada para cada faixa de aprovação?

Monitoramento

- Checar os inputs e outputs do modelo ao longo do tempo.
- As distribuições das variáveis explicativas continuam iguais?
- As relações das variáveis com a resposta continuam iguais?
- A performance continua aceitável?
- As regras de negócio também devem ser monitoradas.

Técnicas

Como combinar todas as possibilidades de modelagem?

Missing

Imputação pela média / mediana
Remover
Imputação por modelo
Multiple Imputation
Categorizar

Outliers

Remover
Truncar por valores mínimos e máximos
Trocar por um modelo
Categorizar

Feature Engineering

Categorização
Splines
Transformações

Feature Selection

Stepwise
mRMR (minimal redundancy maximal relevance)
Random Forest – Importance
SFO (Single Feature Optimisation)
Componentes Principais / Análise Fatorial
Análise de Correlação

Algoritmos

Regressão Linear / Logística
Random Forest
Boosting
Bagging
Redes Neurais

Exemplo

Completed • $5,000 • 925 teams

Give Me Some Credit

Mon 19 Sep 2011 – Thu 15 Dec 2011 (3 years ago)

Dashboard

Home
- Data
- Make a submission

Information
- Description
- Evaluation
- Rules
- Prizes

Forum

Leaderboard
- Public
- Private

My Team
- GitHub

My Submissions

Competition Details » Get the Data » Make a submission

Improve on the state of the art in credit scoring by predicting the probability that somebody will experience financial distress in the next two years.

Banks play a crucial role in market economies. They decide who can get finance and on what terms and can make or break investment decisions. For markets and society to function, individuals and companies need access to credit.

Credit scoring algorithms, which make a guess at the probability of default, are the method banks use to determine whether or not a loan should be granted. This competition requires participants to improve on the state of the art in credit scoring, by predicting the probability that somebody will experience financial distress in the next two years.

Métrica utilizada: AUC

Exemplo

#	Δ1w	Team Name * in the money	Score	Entries	Last Submission UTC (Best – Last Submission)
1	↑47	vsu *	0.863904	14	Thu, 15 Dec 2011 14:02:51
2	↓1	Perfect Storm *	0.863706	128	Thu, 15 Dec 2011 05:35:00 (-21.1h)
3	↑34	Soil *	0.863642	92	Thu, 15 Dec 2011 05:13:12 (-26.7h)
4	↑32	Indy Actuaries	0.863571	23	Thu, 15 Dec 2011 18:35:33 (-3.8d)
5	↓3	SirGuessalot	0.863499	41	Thu, 15 Dec 2011 05:33:10 (-22.5h)
6	↓3	Gxav	0.863356	54	Thu, 15 Dec 2011 09:41:23 (-17.2d)
7	↓2	Xooma	0.863324	74	Thu, 15 Dec 2011 23:25:53 (-45.2h)
8	↓4	Opera Solutions	0.863293	46	Thu, 15 Dec 2011 23:38:23 (-2.5h)
9	↑4	Jason Karpeles	0.863182	70	Thu, 15 Dec 2011 22:39:26 (-2.8d)
10	↑67	Winter is Coming	0.863046	10	Thu, 15 Dec 2011 22:22:54
11	↓5	UCI-CS273a-FabSadBac	0.862959	52	Sat, 10 Dec 2011 22:58:06 (-6d)
12	↓5	StephenYe	0.862903	18	Thu, 15 Dec 2011 16:33:01 (-6.8d)
13	↓5	B Yang	0.862880	65	Thu, 15 Dec 2011 23:43:58 (-14d)
14	↑95	lucky guy	0.862840	16	Thu, 15 Dec 2011 17:48:21 (-4.5d)