Python & Machine Learning no desenvolvimento de tecnologias para a indústria: um caso prático

Unisinos Python Day 2018

Ariel Kempf

TÓPICOS

- Python, Indústria e Machine Learning
- Trisolutions & Python
- O problema
- Metodologia
- Resultados
- Conclusões e trabalhos futuros
- Bibliografia e agradecimentos

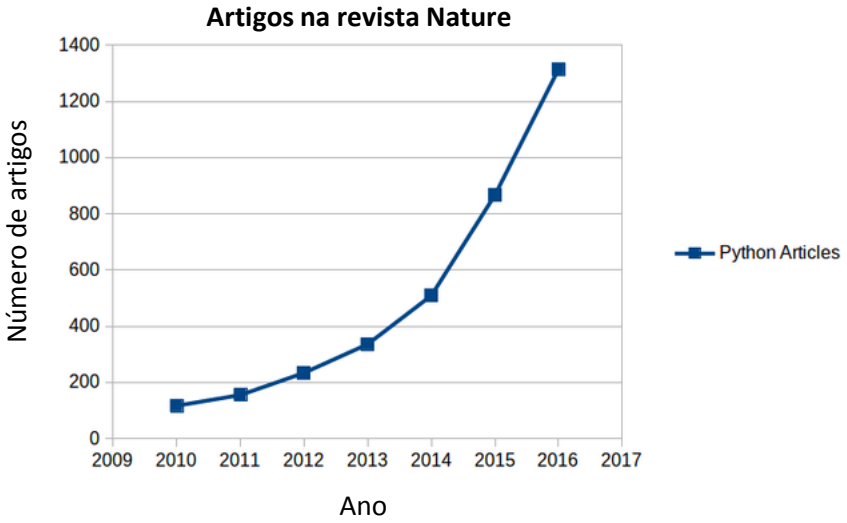
Python, Indústria e Machine Learning

TRISOLUTIONS®

Python

Python se tornou bastante popular no meio acadêmico:

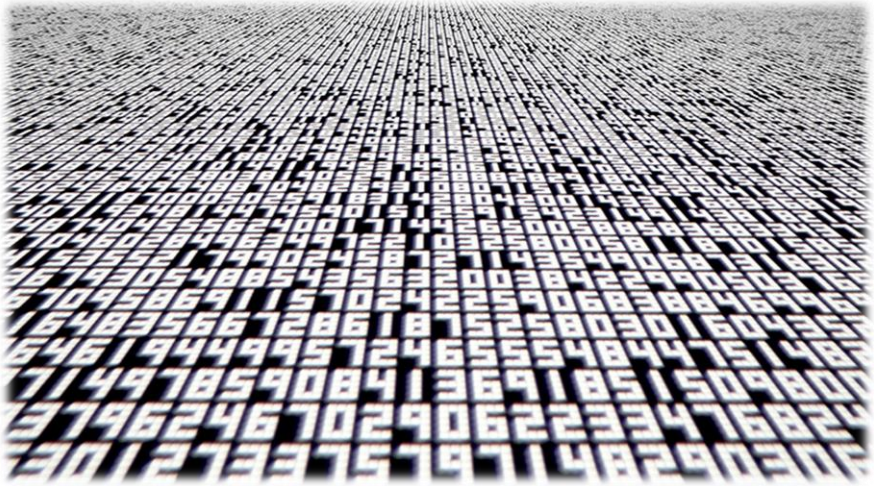
- Estruturas para lidar eficientemente com grandes quantidades de dados
- Simples e rápida de aprender
- Gratuito e livre de royalties
- Ótimas bibliotecas
- https://www.python.org/



https://www.quora.com/Why-has-Python-become-so-popular-in-academia-superseding-other-languages-like-C-C++-Java-and-C

Indústria

http://www.ryojiikeda.com/project/datamatics/



Indústria de transformação contínua altamente automatizada:

- Automatização → instrumentação
- Ordem de dezenas de milhares de variáveis
  - Controle
  - Proteção/segurança/meio-ambiente
  - Parametrização do processo
- Dados brutos armazenados por PIMS
- Usos óbvios como monitoração visual, estatísticas descritivas simples

Machine Learning

Conjunto de técnicas que visam ensinar computadores a aprender com a experiência

Existentes há bastante tempo

Só recentemente vem se popularizando:

- Melhorias nos métodos estatísticos propriamente ditos
- Aumento da capacidade computacional para trabalhar com grandes conjuntos de dados
- Disponibilidade de grandes capacidades de dados
- Popularização de bibliotecas de programação gratuitas e de código aberto

"Área de conhecimento que visa conferir aos computadores a habilidade de aprender sem serem explicitamente programados."

Arthur Samuel (1959)

"Um programa de computador é considerado estar *aprendendo* a partir de uma experiência E com respeito a alguma tarefa T e um desempenho D se seu desempenho em T, tal como medido por D, melhora com a experiência E."

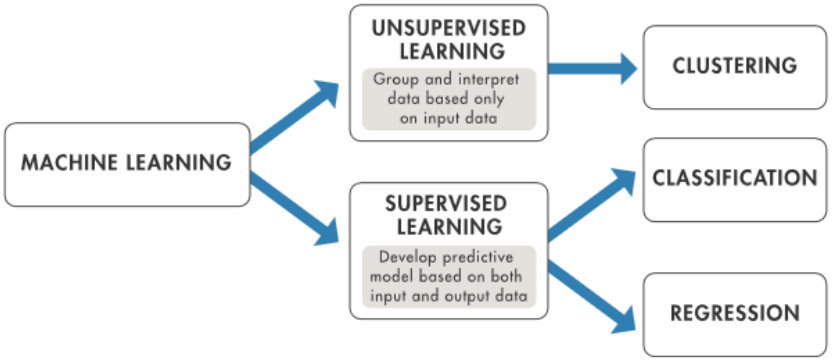
Tom Mitchell (1998)

Machine Learning

Normalmente classificados em duas categorias amplas:

- Aprendizado supervisionado
  - Regressão
  - Classificação
- Aprendizado não-supervisionado
  - *Clustering*

- Curso:
  https://www.coursera.org/learn/machine-learning
  Andrew Ng
  Universidade de Stanford



Usos:

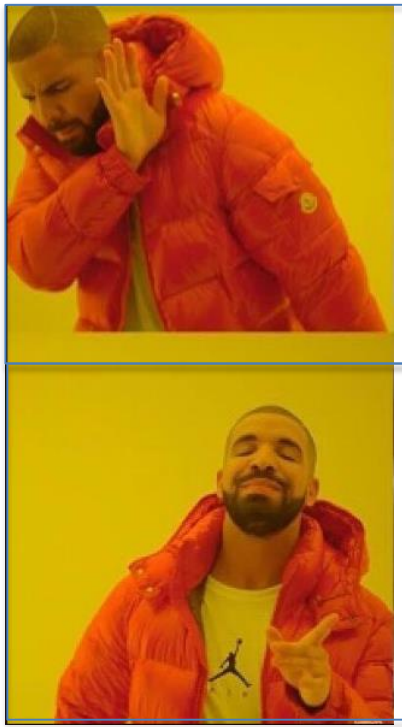
- Mineração de dados
- Detecção de padrões, falhas, anomalias
- Reconhecimento de caligrafia
- Redução de dimensionalidade
- Visão por computador
- Reconhecimento facial
- Filtros de spam
- Processamento de Linguagem Natural
- Recomendação de produtos
- Manutenção preditiva, etc.

Machine Learning

Machine Learning + Python

Permite que se “desbloqueie” um grande potencial de uso de dados brutos subaproveitados:

- Estruturas básicas de dados: listas, mapas, tuplas
- Estruturas avançadas: Numpy, Pandas
  - www.numpy.org/
  - https://pandas.pydata.org/
- Visualização: Matplotlib
  - https://matplotlib.org/
- Machine Learning: Scikit-learn
  - http://scikit-learn.org/



- **Big data**
- **Inspeção visual**
- **Estatísticas simples**

- **Big data**
- **Python**
- **Machine learning**

Trisolutions & Python

TRISOLUTIONS®

A PGA

PGA - Plataforma de Gestão de Ativos

Prêmio ANP de Inovação Tecnológica – 2017

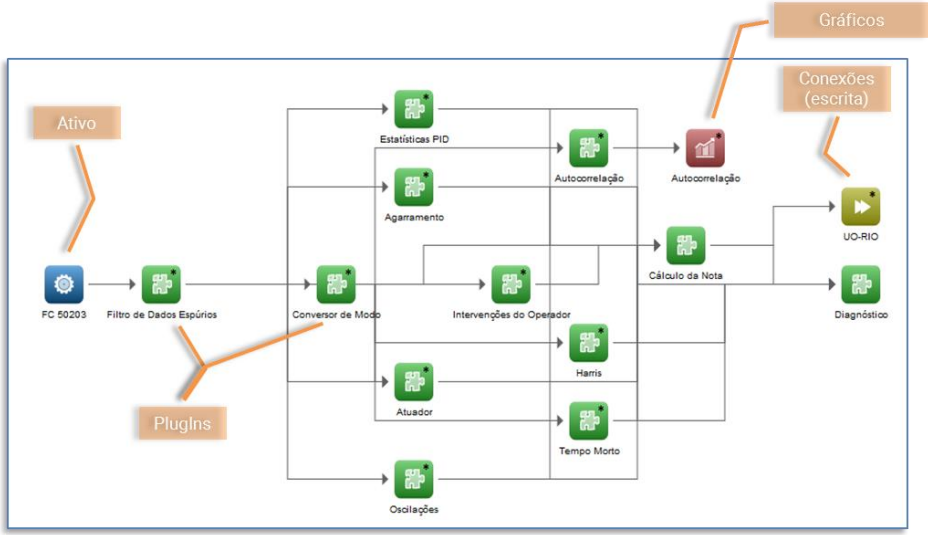
1º Lugar – Categoria II
Petrobras/CENPES
UFRGS
TRISOLUTIONS

“Pilotos de Sistemas de Manutenção Preditiva no SSE”

Trisolutions & Python

A PGA

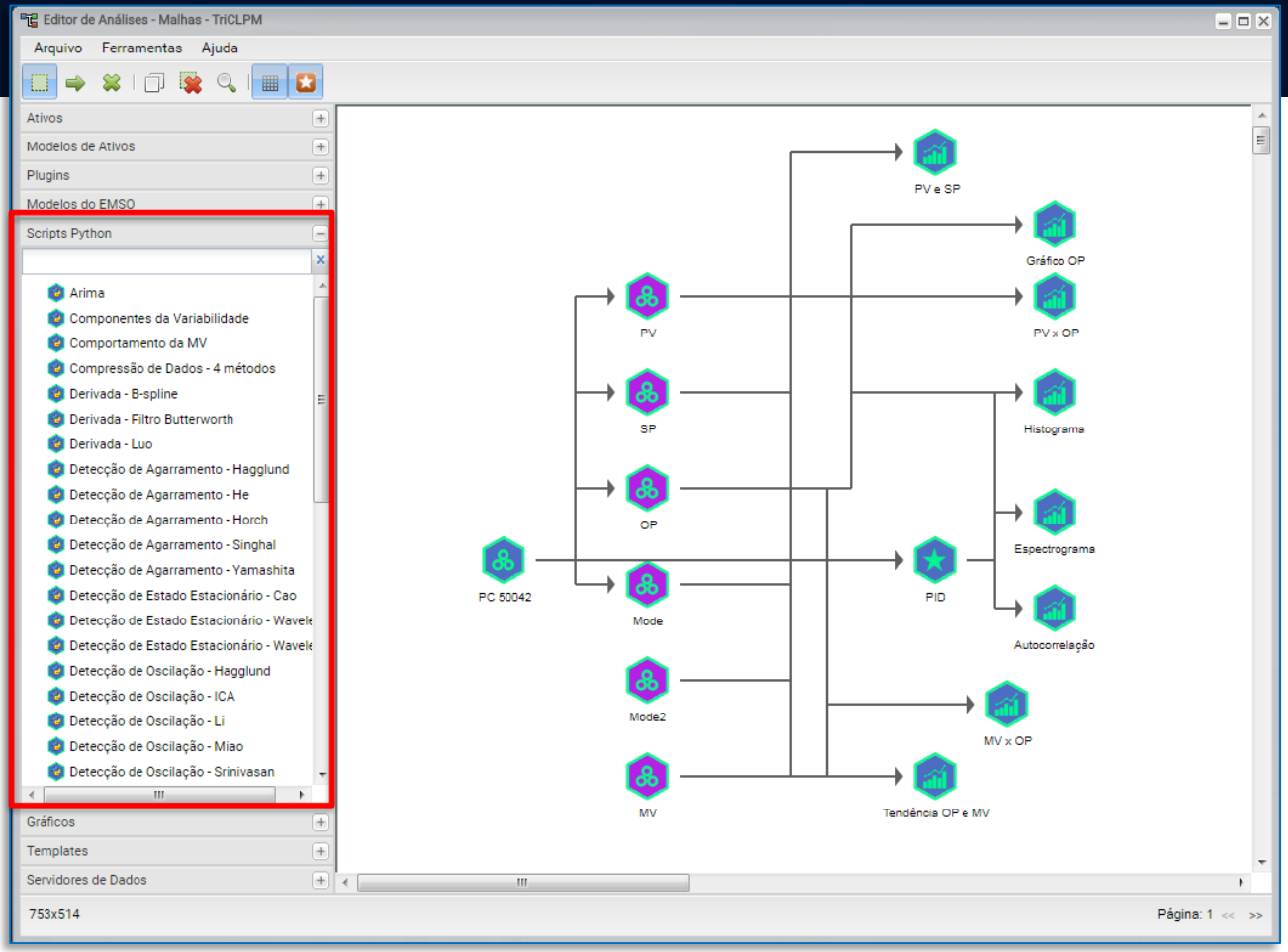
- Acesso às fontes de dados industriais mais comuns
- Biblioteca de blocos de cálculo para processamento de sinais, índices de desempenho, condição de equipamentos, pré-tratamento, etc.
- Visualização totalmente WEB
- Dashboards
- Análises



Trisolutions & Python

A PGA

Integração Python

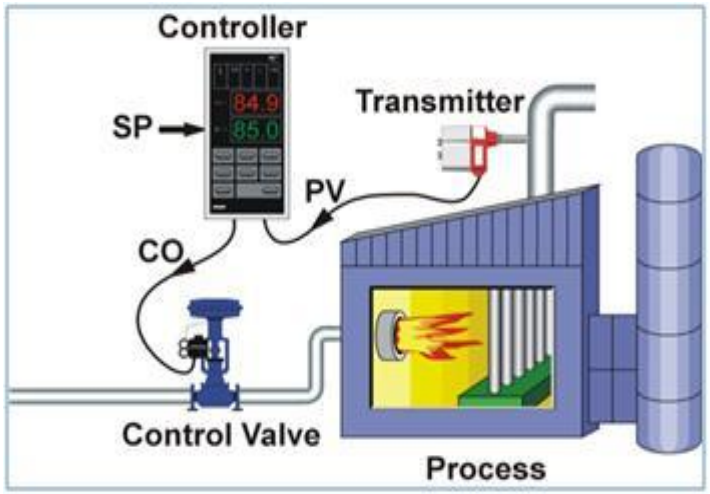


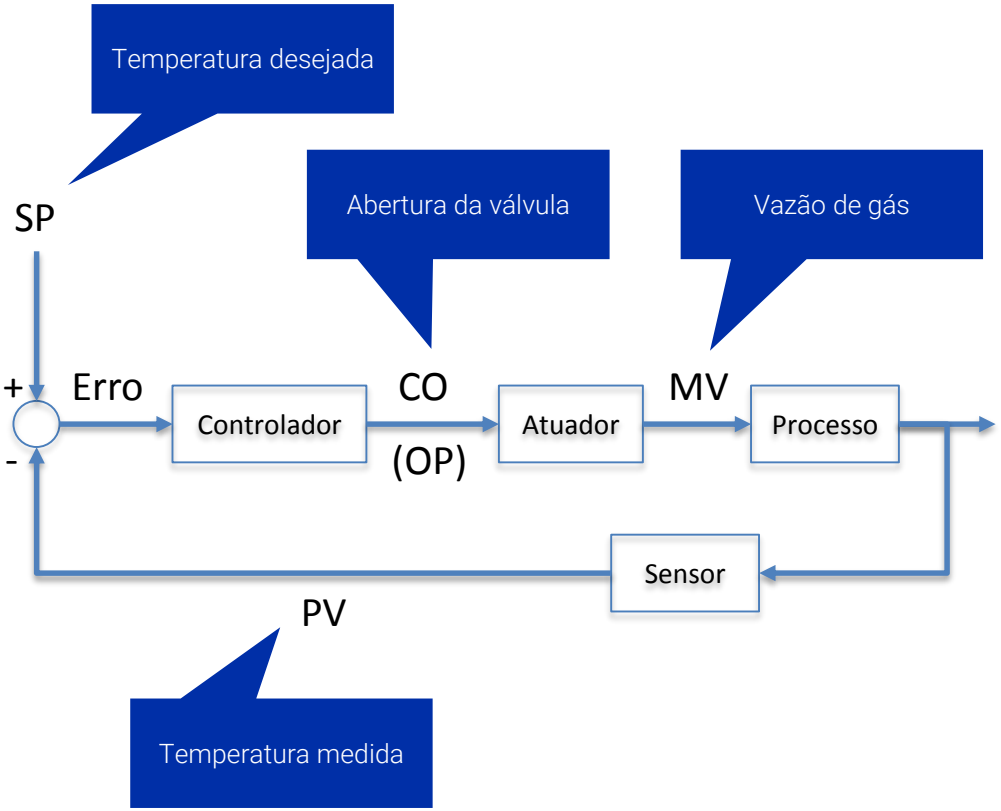
O Problema

O problema

Controle automático

Controle baseado em *feedback*





O problema

Controle automático

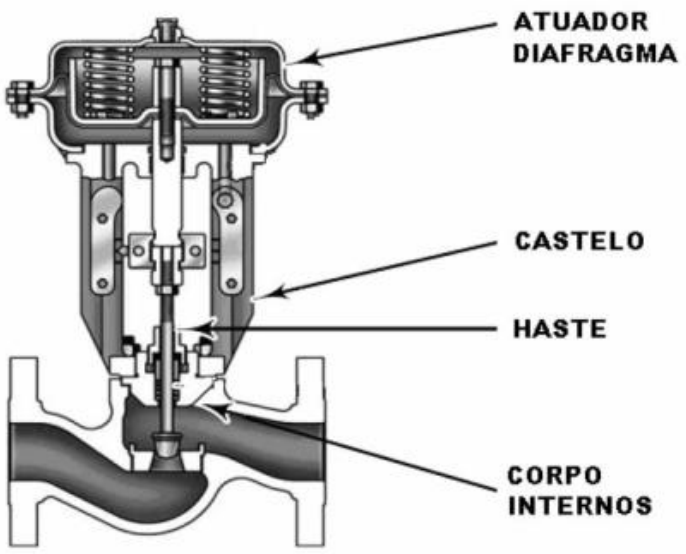
Controle baseado em *feedback* – Problemas que podem ocorrer:

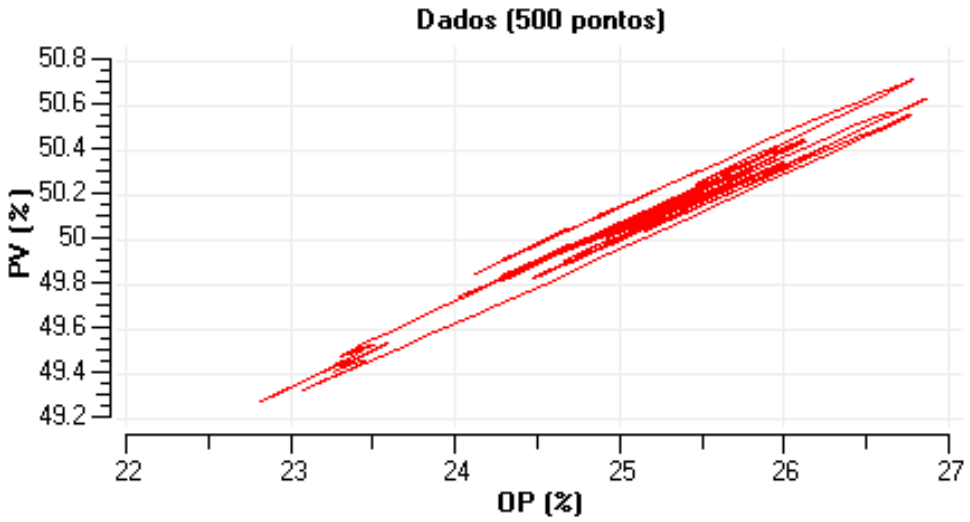
- Problemas na parametrização do controlador
- Problemas causados por distúrbios externos e mudanças na dinâmica
- Problemas envolvendo o sensor
- Problemas envolvendo o atuador

O problema

Válvulas de controle

- Funcionamento desejado

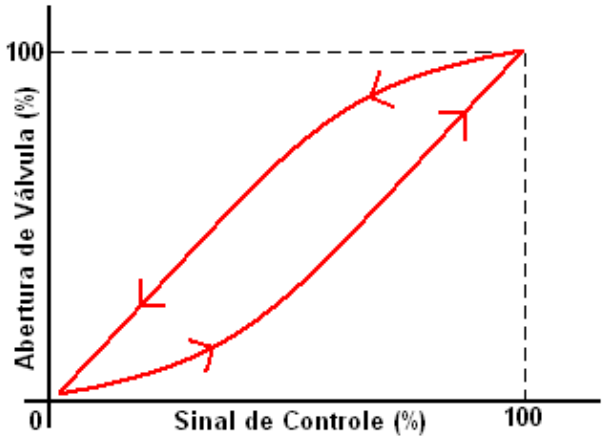


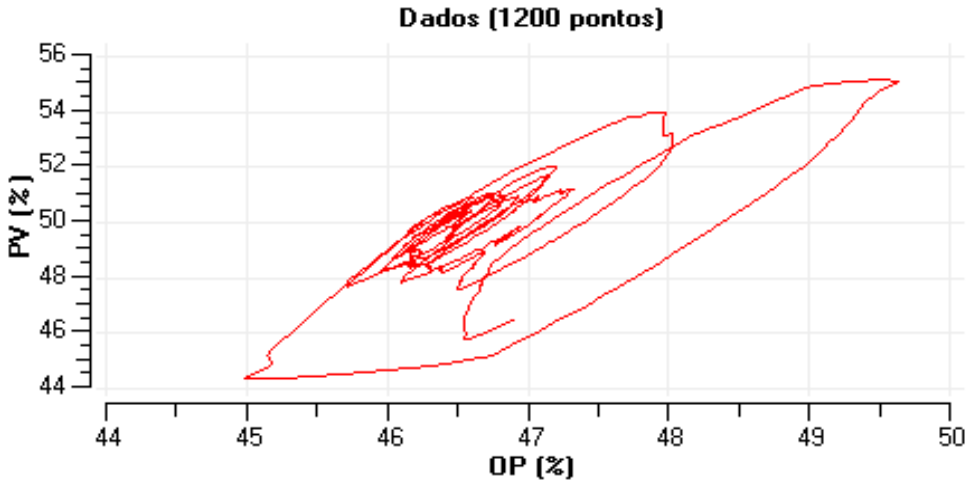


O problema

Válvulas de controle

- Histerese – atrito dinâmico ao longo do curso da válvula

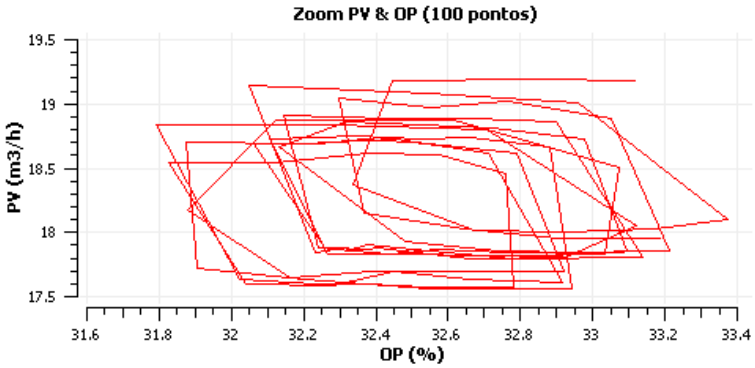


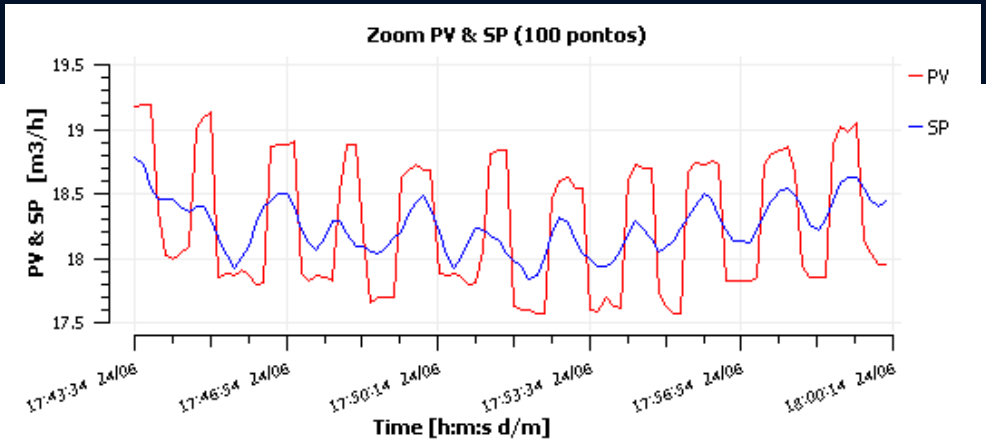


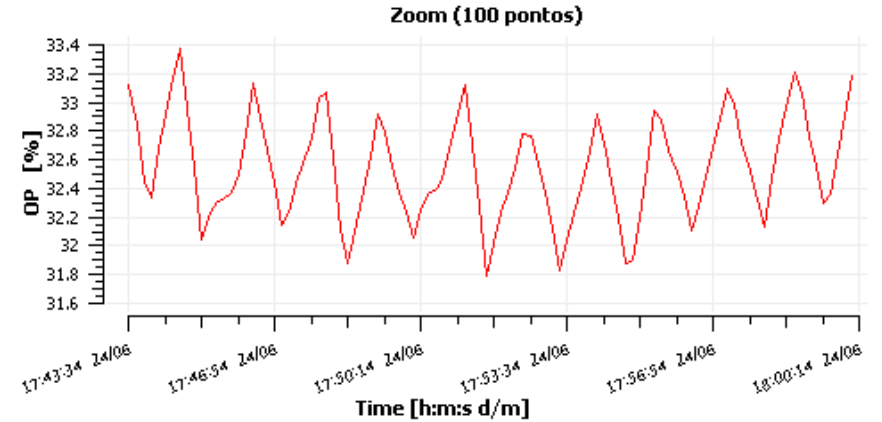
O problema

Válvulas de controle

- Agarramento – Stiction
- (Static Friction), atrito estático







O problema

Agarramento

Problema considerável para o desempenho da malha de controle e do processo

- Comportamento oscilatório da variável controlada
- Propaga-se para outras partes do processo
- 30% do comportamento oscilatório é devido ao agarramento [Bialkowski, 1992]
- Malhas com baixo desempenho afetam o retorno financeiro da unidade produtiva

O problema

Objetivos

Objetivo das equipes de automação, processo e manutenção:

Como encontrar, dentro das centenas de válvulas de controle no processo, quais apresentam agarramento?

Objetivo da TRISOLUTIONS:

Desenvolver uma metodologia com a finalidade de detectar malhas com agarramento e apresentar o diagnóstico para os usuários do PGA

Metodologia

TRISOLUTIONS®

Metodologia

Premissas

- Posição real da válvula:
    - Pode ser medida por um posicionador inteligente integrado à válvula
    - Pode ser uma informação que retorna ao sistema de controle e está disponível

Premissas:

- Apenas dados de PV, SP e OP disponíveis
- Retorno do posicionador não disponível no sistema de controle
    - Pode dar resultados melhores se usado onde disponível
    - A mesma metodologia baseada em ML pode ser utilizada

Metodologia

Revisão

Diversas técnicas não-invasivas para identificação [Dambros, 2016]

Baseadas no padrão PV, OP ou ambas.

- [Horch, 1999]
- [Yamashita, 2006]
- [Singhal, 2005]
- [He, 2007]
- [Choudhury, 2008]

Algumas podem trabalhar com retorno do posicionador

[Dambros, 2016] fornece uma revisão com as limitações destes métodos

Metodologia

Machine Learning

Métodos de aprendizado supervisionado para classificação

Dados *m* exemplos de treinamento na forma $\{(x_1, y_1), \ldots, (x_m, y_m)\}$

Encontre os parâmetros da função $h(\theta)$: X → Y

Que minimizam uma função-custo baseada em alguma métrica, $J(\theta)$

Metodologia

Machine Learning

Métodos de aprendizado supervisionado para classificação
Exemplo: detecção de câncer

- Atributos de entrada (Xi)
  - Plaquetas
  - Fumante?
  - Caso na família?
  - Colesterol
  - Enzima específica no sangue?
  - Nível de hormônio
  - ...

- Rótulos (Yi)
  - Negativo (0)
  - Positivo (1)

Metodologia

Machine Learning

Métodos de aprendizado supervisionado para classificação

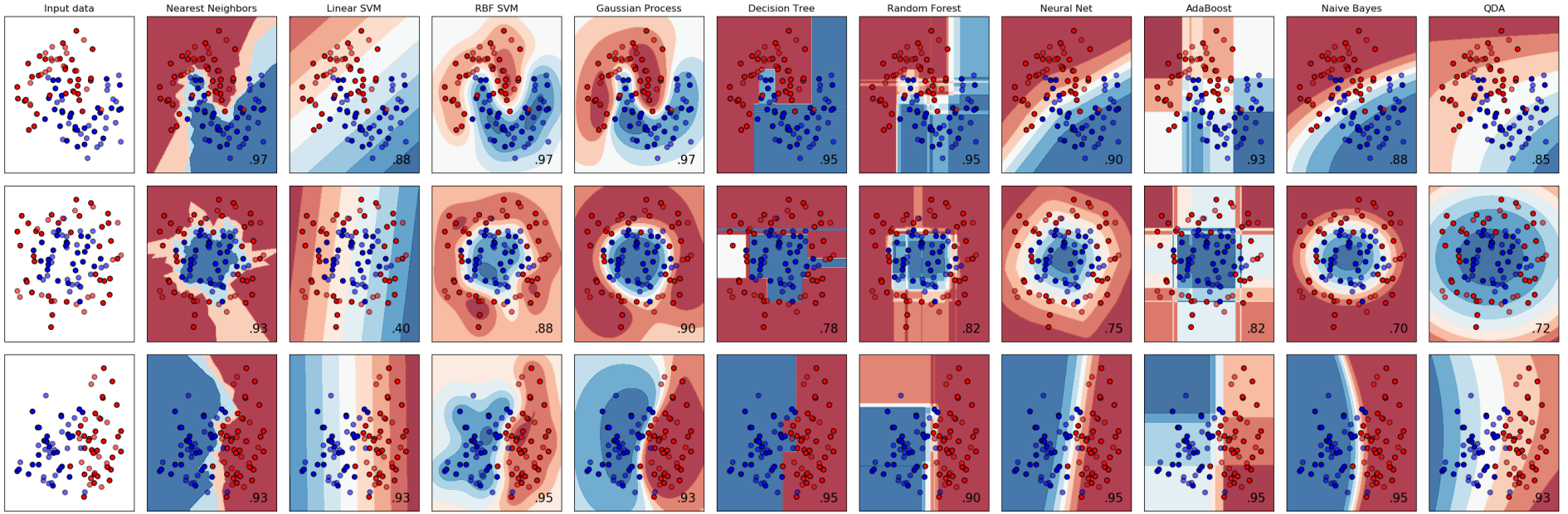
Algoritmos:

- Regressão logística
- Suport Vector Machine (SVM)
- Redes neurais artificiais (ANN)
- K-Nearest Neighbor (KNN)
- Árvores de decisão
- Processo Gaussiano

http://scikit-learn.org/stable/auto_examples/classification/plot_classifier_comparison.html

Metodologia

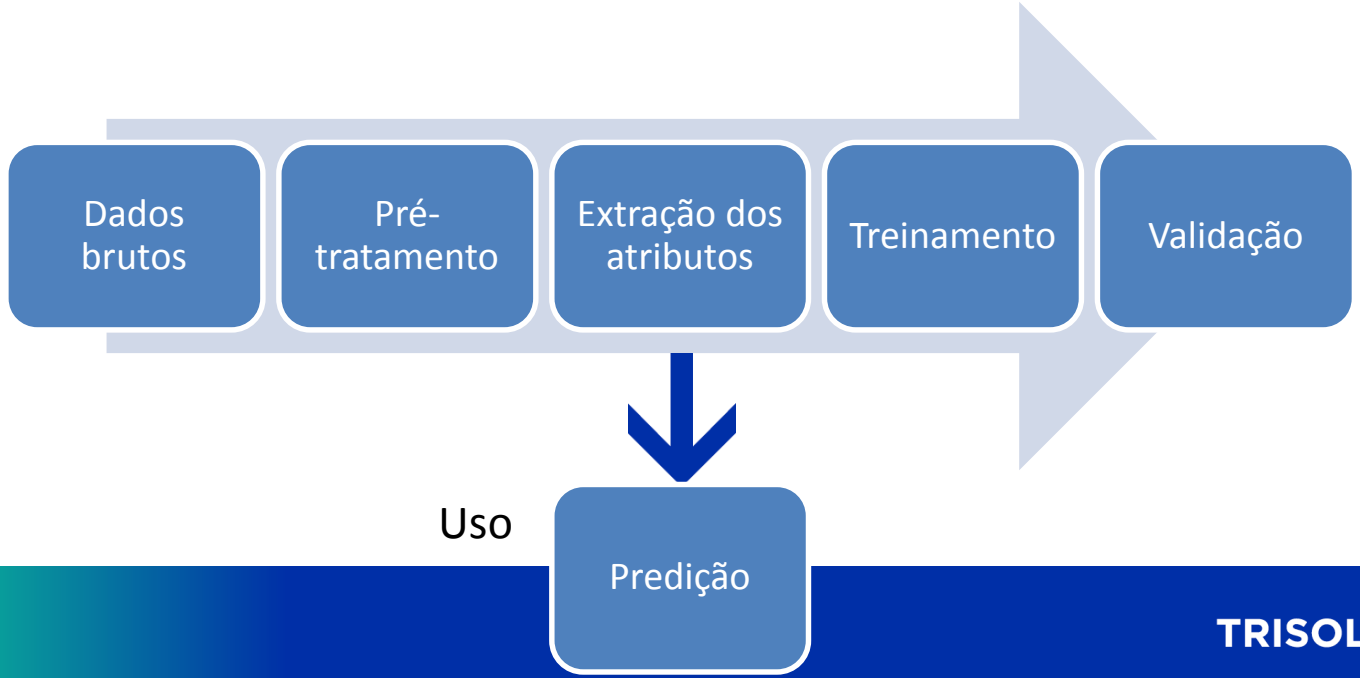
Machine Learning



Metodologia

Machine Learning

PIPELINE: processo de obtenção do modelo de classificação e uso



Machine Learning

Primeiro grande segredo do Machine Learning:

Que atributos dos dados usar em X para chegar em Y?

Metodologia

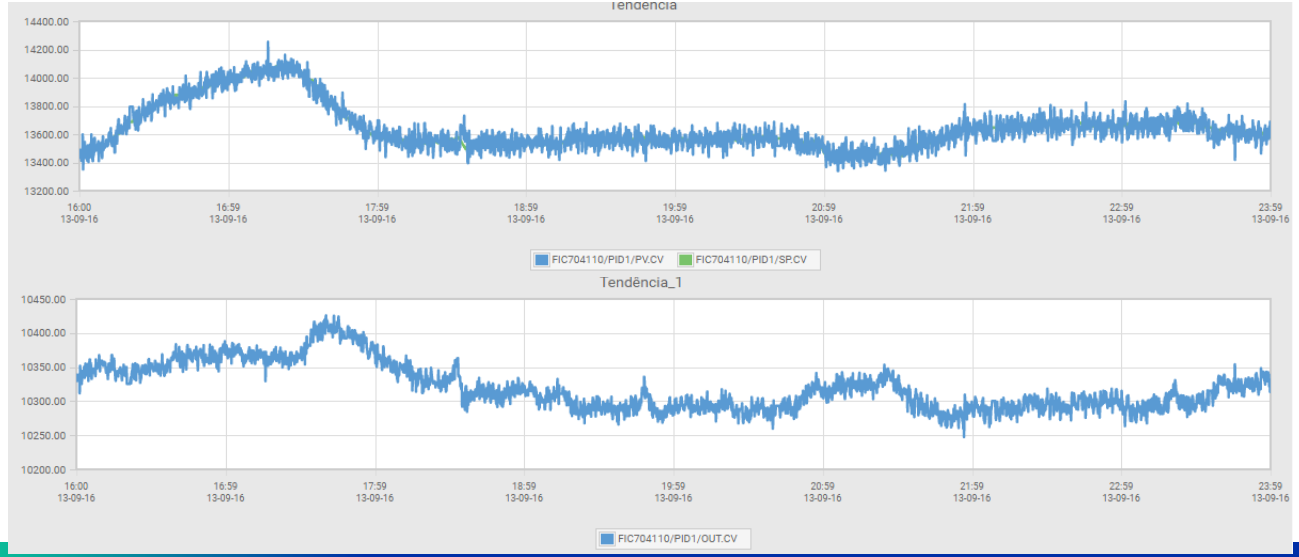
Detecção de agarramento

Dados brutos: não posso usar diretamente como X

- Alta dimensionalidade

PV

OP



Sim (1)

Não (0)

PV + OP = 2n atributos!

TRISOLUTIONS®

Extração dos atributos

Usar estatísticas descritivas?

- O meio-acadêmico já havia tentado

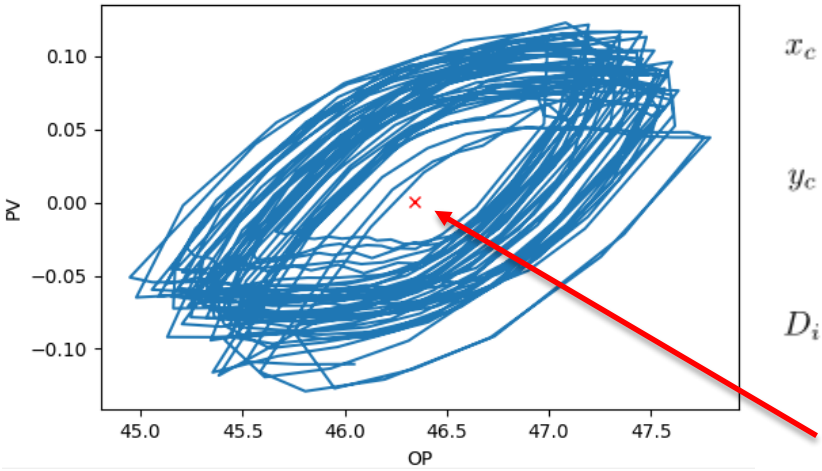
  - Centro – média/mediana/modo
  - Dispersão – desvio-padrão, faixa, percentil
  - Forma – curtose, assimetria

Metodologia

Extração dos atributos

Bibliografia: [Venceslau, Guedes, Silva, 2012]

Proposta de redução de dimensionalidade: distância para o centróide PV x OP



$$x_c = \frac{1}{N}\sum_{i=1}^{N} x_i$$

$$y_c = \frac{1}{N}\sum_{i=1}^{N} y_i$$

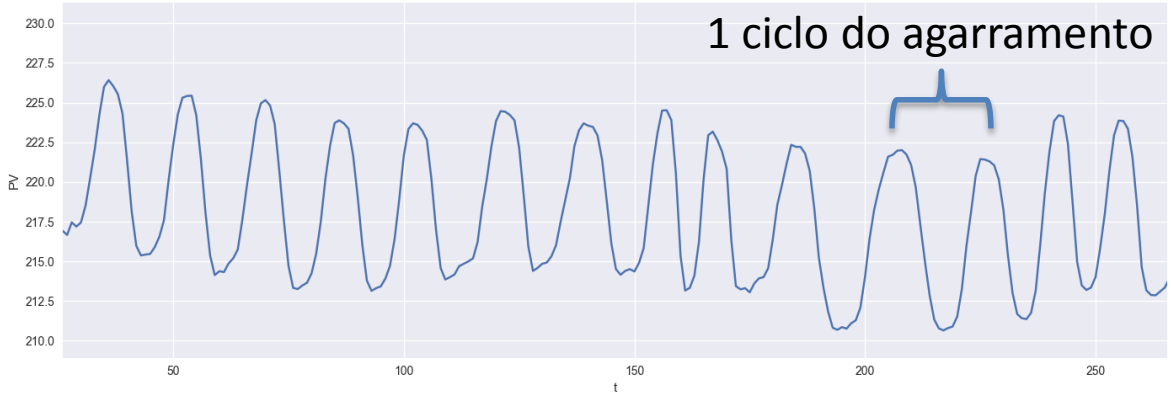
$$D_i = \sqrt{(x_i - x_c)^2 + (y_i - y_c)^2}$$

2n → n atributos

Extração dos atributos

Mas **n** ainda é uma dimensão muito alta!

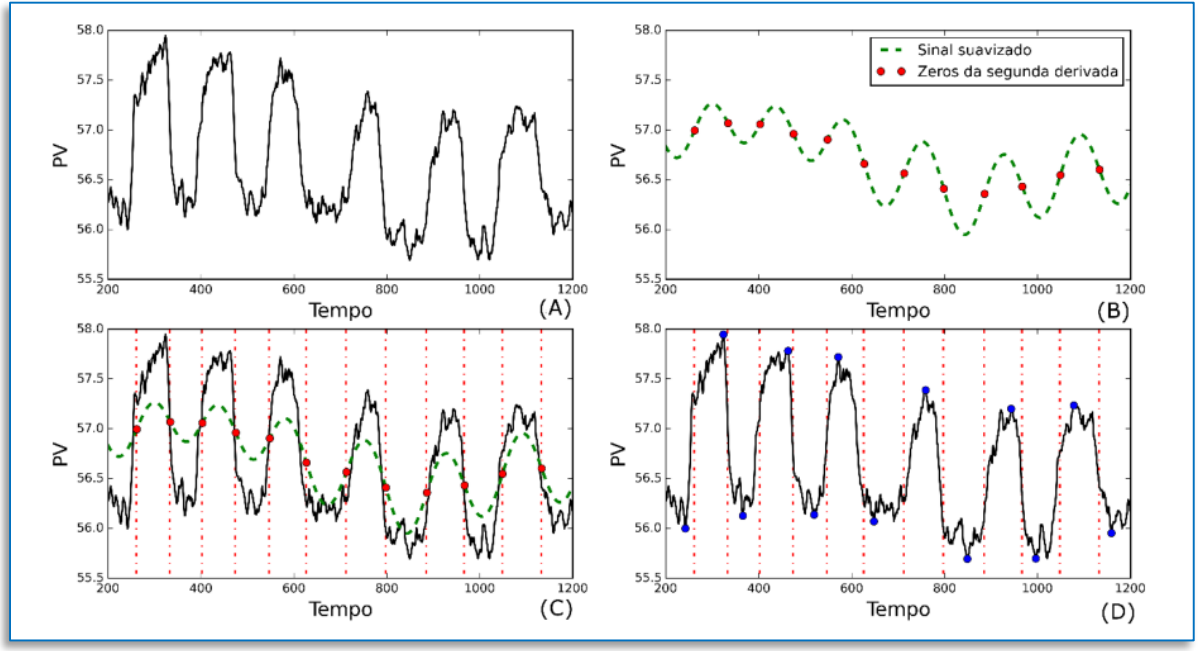
E se eu usar alguns ciclos do agarramento?



Mas como detectar e isolar ciclos de agarramento?

Metodologia

Extração dos atributos

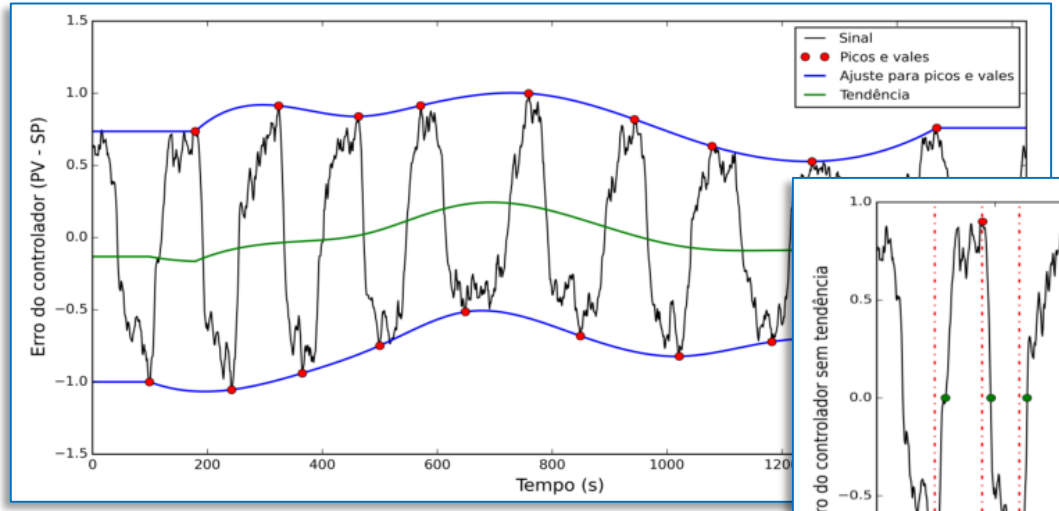


[Dambros, 2016] propõe método de remoção de tendência para sinais de malhas com setpoint (SP) variável

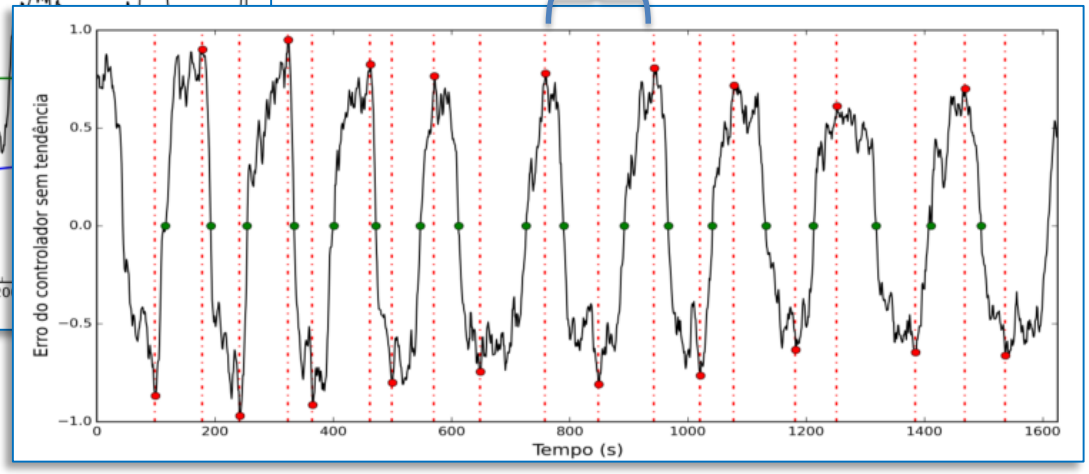
Metodologia

Extração dos atributos

[Dambros, 2016]: método de remoção de tendência



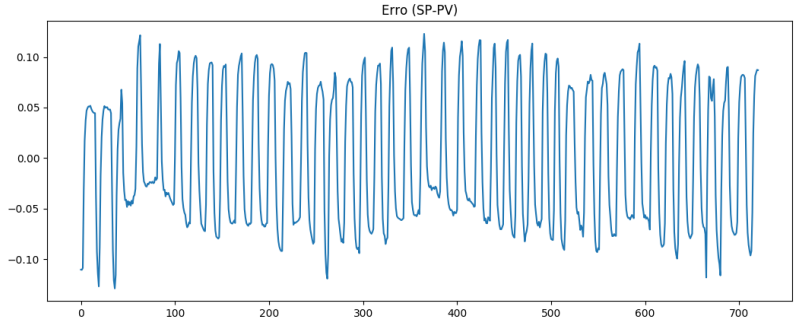
1 ciclo do agarramento

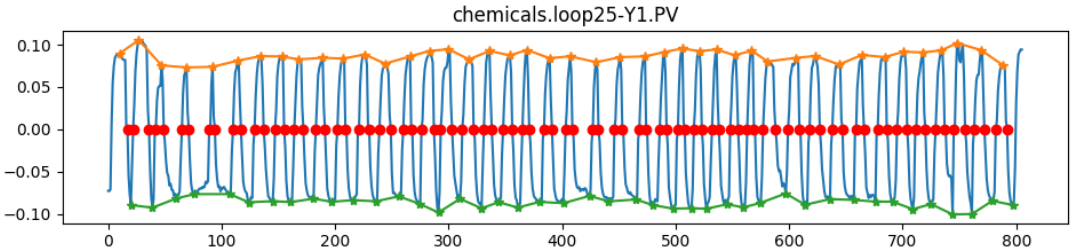


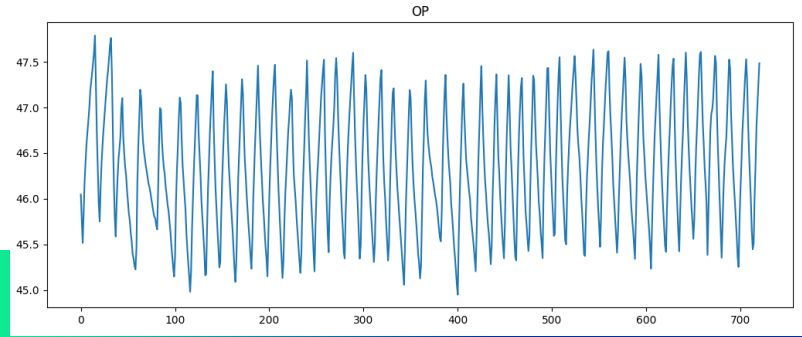
Metodologia

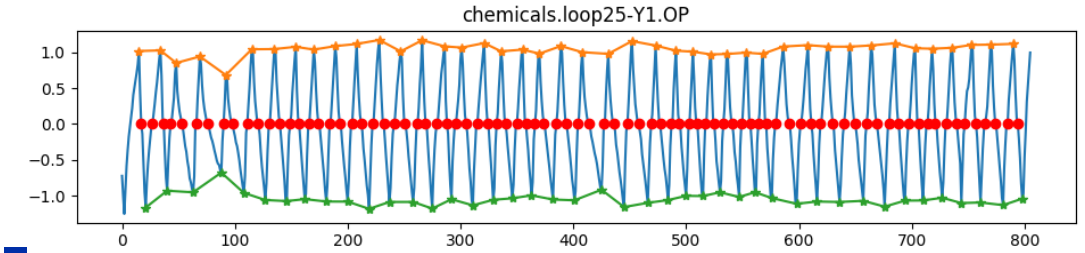
Extração dos atributos

Aplicando a técnica: encontra segmentos na PV (erro) e aplica na OP









TRISOLUTIONS®

Metodologia

Extração dos atributos

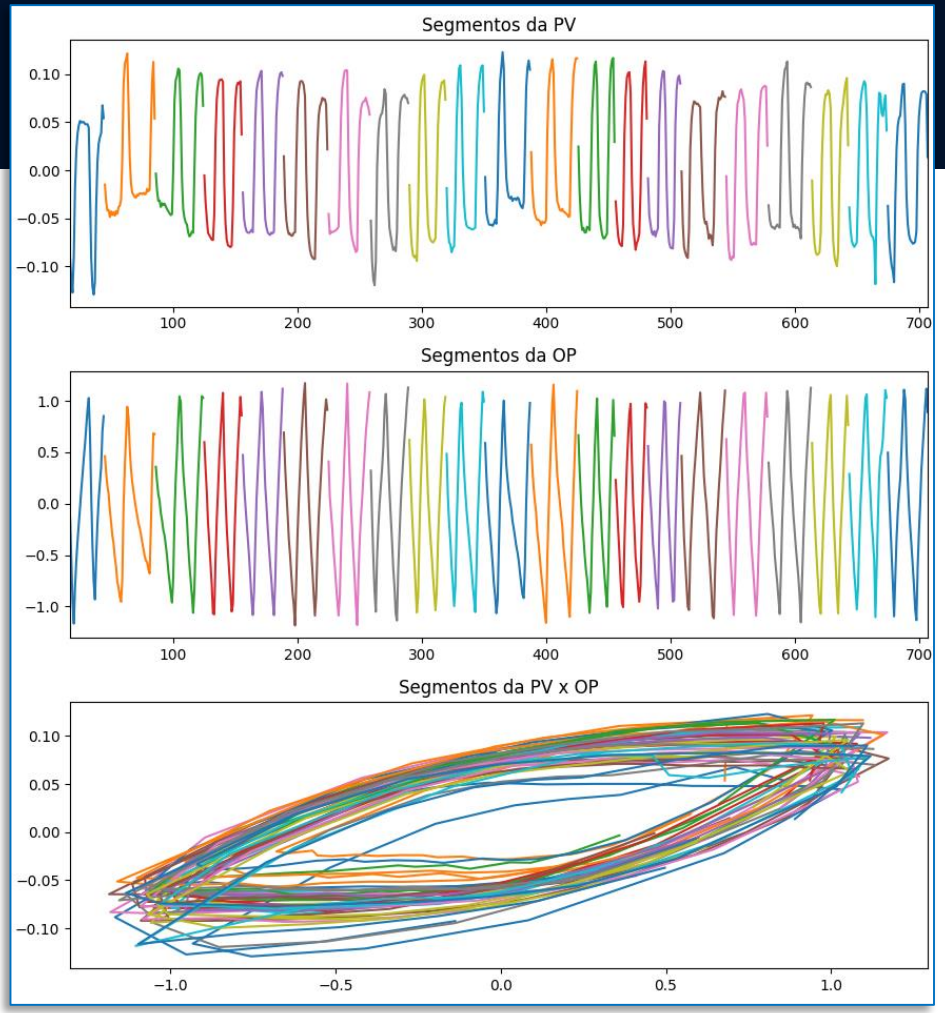
Considerando:

um segmento = 4 ciclos de agarramento:

Ao invés de usar n pontos das distâncias ao centroide, vamos usar um segmento médio ou mais comum.

Mas ainda temos um problema:

o tempo de amostragem dos dados



Extração dos atributos

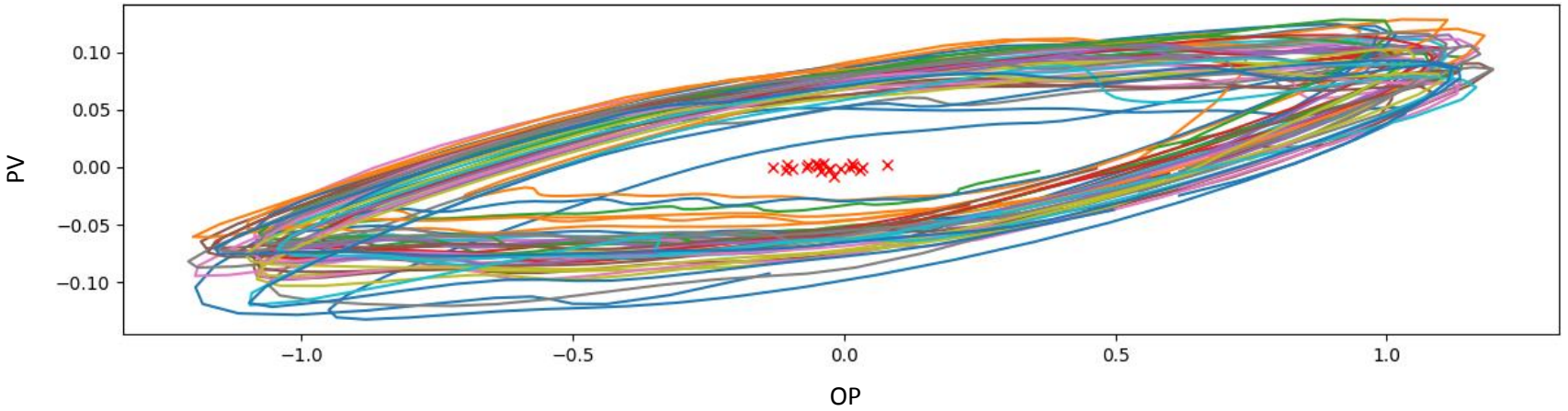
O tempo de amostragem:

- Modelo de classificação com ML precisa de número fixo de atributos [X = O(n)]
- Diferentes tempos de amostragem produzem ciclos com número de pontos diferentes
- É necessário reamostrar segmentos para ficarem sempre com mesmo tamanho (`scipy.interpolate.interp1d`)
- Escolha: 120 pontos por segmento de 4 ciclos

Extração dos atributos

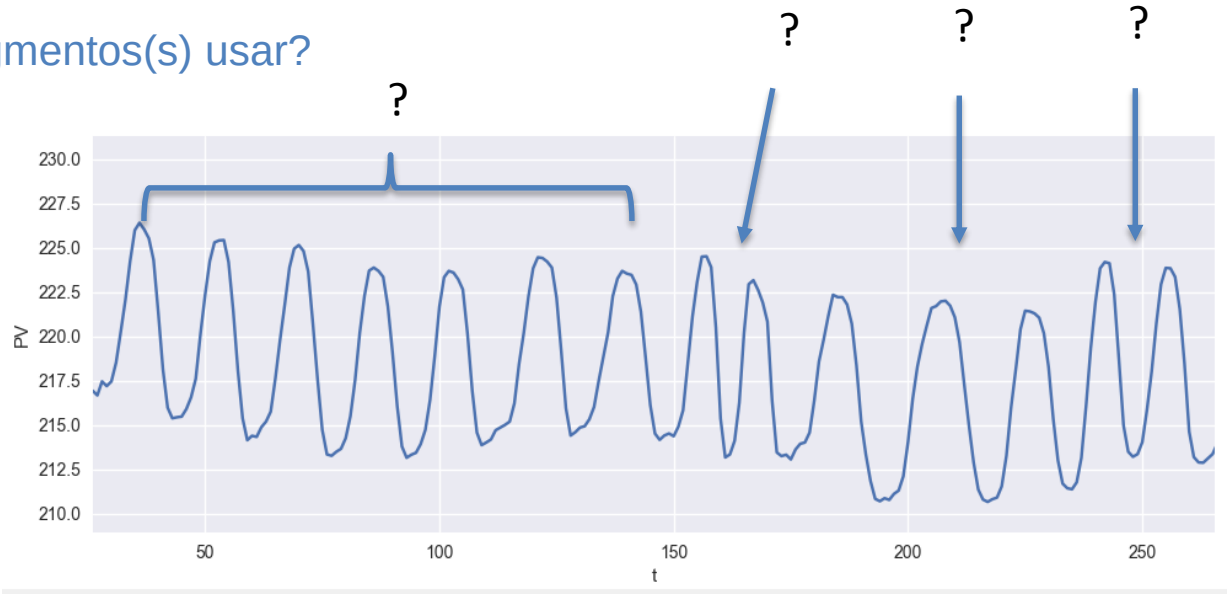
Agora pode-se calcular as distâncias para o centroide.

Cada segmento tem um centroide



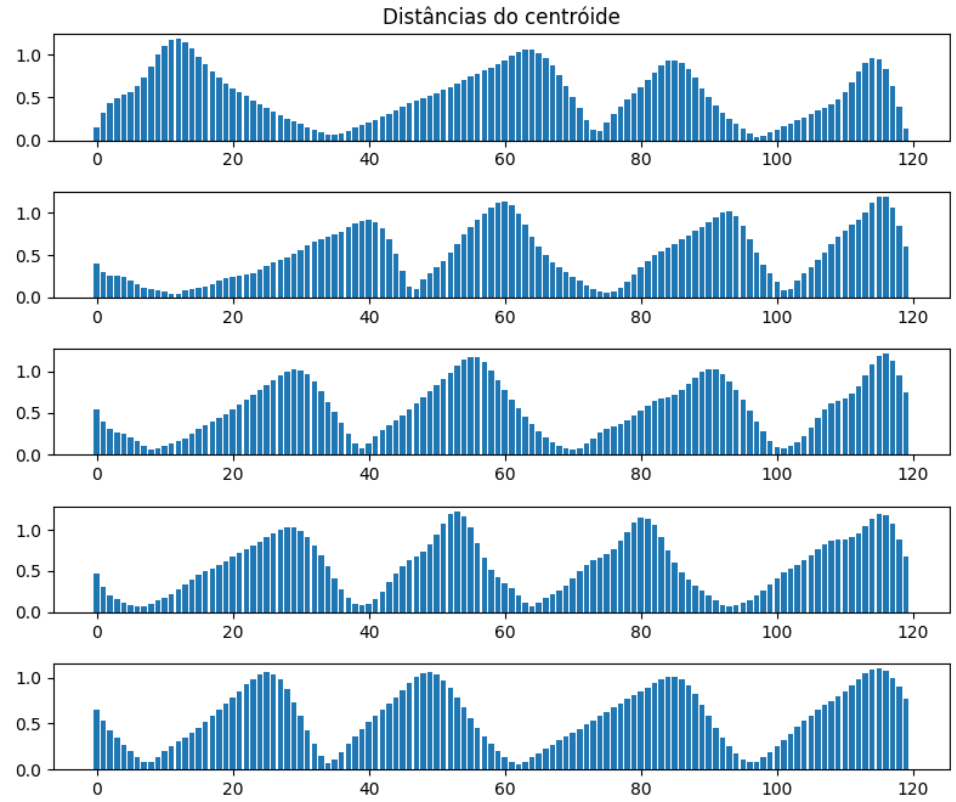
Extração dos atributos

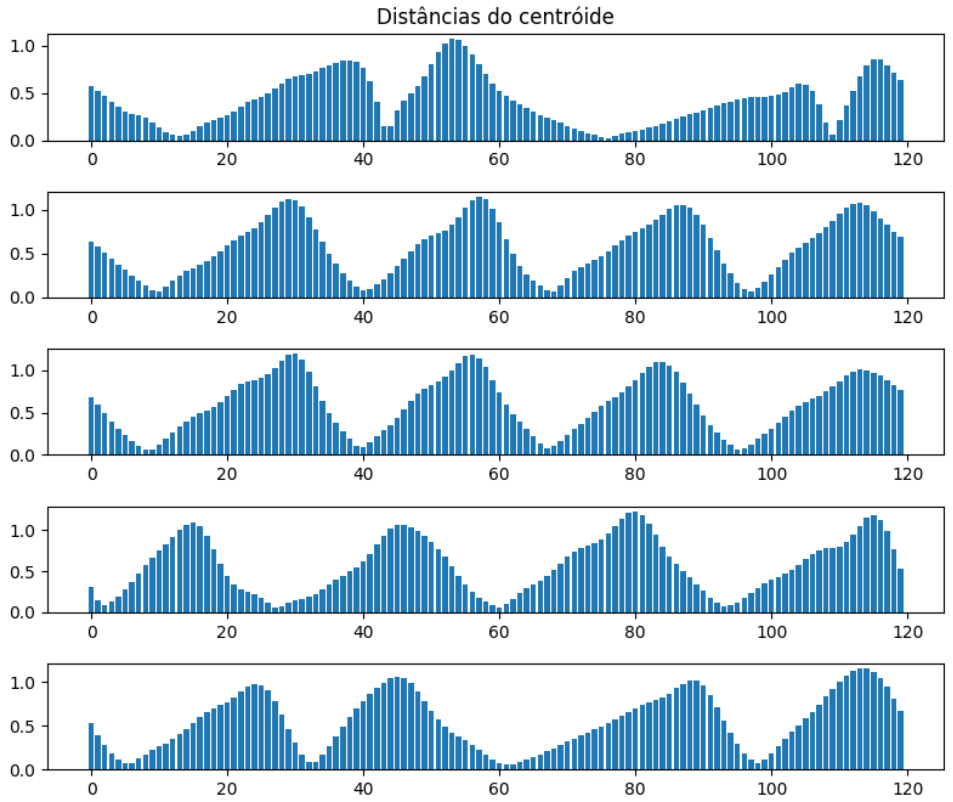
Qual(is) segmentos(s) usar?



Metodologia

Extração dos atributos





Extração dos atributos

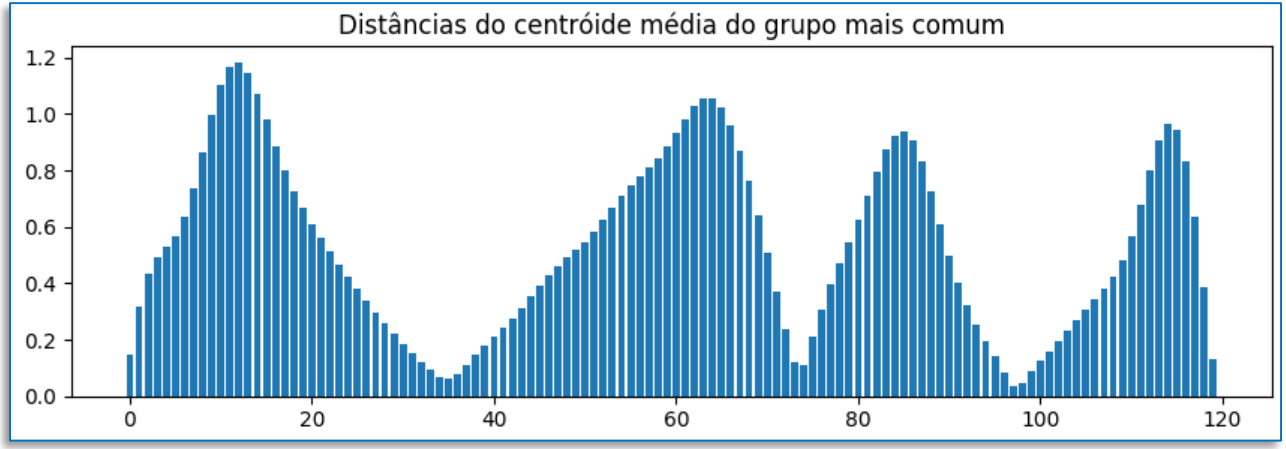
Qual(is) segmentos(s) usar?

Proposta: segmento médio do grupo de segmentos mais comum nos dados

- Aplica outro algoritmo de Machine Learning: K-means clustering (`sklearn.cluster.Kmeans` e `skearn.metrics.silhouette_score`)
- Vai agrupar os segmentos por similaridade nas distâncias ao centroide entre si
- O grupo com mais segmentos (mais comum) é escolhido e um segmento médio é calculado

Extração dos atributos

Segmento médio do grupo com mais segmentos similares entre si



X

Machine Learning

Segundo grande segredo do Machine Learning:

De onde obter dados em quantidade suficiente para o treinamento?

Metodologia

Machine Learning

"It's not who has the best algorithm that wins. It's who has the most data." (Andrew Ng)

Possíveis soluções:

- Buscar mais dados reais
    - Agarramento indústria ~5% petróleo/petroquímica; ~15% mineração/siderurgia
    - Meio acadêmico: International Stiction Data Base (ISDB) [Jelali e Huang, 2010]
- Simular malhas de controle
    - Algoritmo PID em Python: adaptado de ivPID (http://ivmech.github.io/ivPID)
    - Simulação de sistemas de primeira e segunda ordem (`scipy.integrate.odeint`)
    - Simulação de válvulas de controle com e sem agarramento [Choudhury, 2005]

Metodologia

Machine Learning

Simulação:

- Variando:
    - Ganho, integral do controlador
    - Ganho e constante de tempo do modelo do processo
    - Tipo de válvula (linear e igual percentagem)
    - Agarramento e folga da válvula
- Fator de corte para rotulagem (Y) dos dados simulados: if agarramento >= 1%: “Sim” else: “Não”
- Rodando em paralelo com `joblib.Parallel` e `joblib.delayed`

Metodologia

Machine Learning

Simulação, pré-tratamento e adição de malhas reais:

- 13866 simulações executadas
- 7832 simulações sem saturação
- 5513 malhas com segmentos válidos
- 98 malhas reais
    - `sklearn.model_selection.train_test_split`
    - 3927 malhas usadas no treinamento (70%)
    - 1684 malhas usadas na validação cruzada (30%)

Metodologia

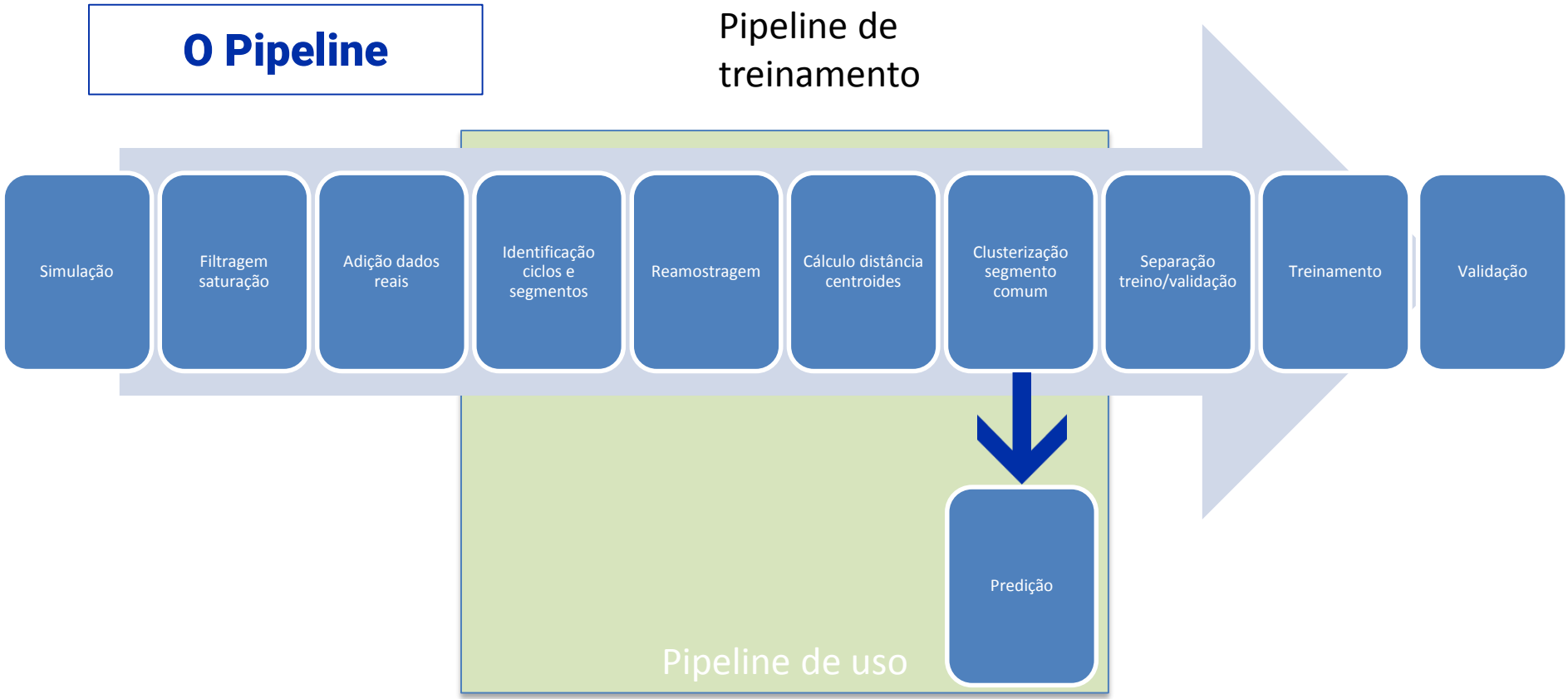
Machine Learning

Treinamento:

- Pesos dos dados de malhas reais: 10000:1
- Testados modelos de classificação
    - Rede neuronal artificial: `sklearn.neural_network.MLPClassifier`
    - Regressão logística: `sklearn.linear_model.LogisticRegression`
    - Nearest neighbor: `sklearn.neighbors.KNeighborsClassifier`
    - SVM com kernel RBF (radial basis function): `sklearn.svm.SVC`

Metodologia

O Pipeline



Resultados

TRISOLUTIONS®

Desempenho do classificador

Como saber se o modelo está bom?

- Avaliar:
  - Falsos positivos (FP)
  - Falsos negativos (FN)
  - Verdadeiros positivos (VP)
  - Verdadeiros negativos (VN)

		Classe real	
		1	0
Classe predita	1	Verdadeiro positivo	Falso positivo
	0	Falso Negativo	Verdadeiro Negativo

Desempenho do classificador

Como saber se o modelo está bom?

- Acurácia:

$$Acurácia = \frac{VP + VN}{VP + VN + FP + FN}$$

		Classe real	
		1	0
Classe predita	1	VP	FP
	0	FN	VN

```
sklearn.metrics.accuracy_score()
```

Desempenho do classificador

Como saber se o modelo está bom?

- Precisão e Recall:

$$Precisão = \frac{VP}{VP + FP}$$

$$Recall = \frac{VP}{VP + FN}$$

		Classe real	
		1	0
Classe predita	1	VP	FP
	0	FN	VN

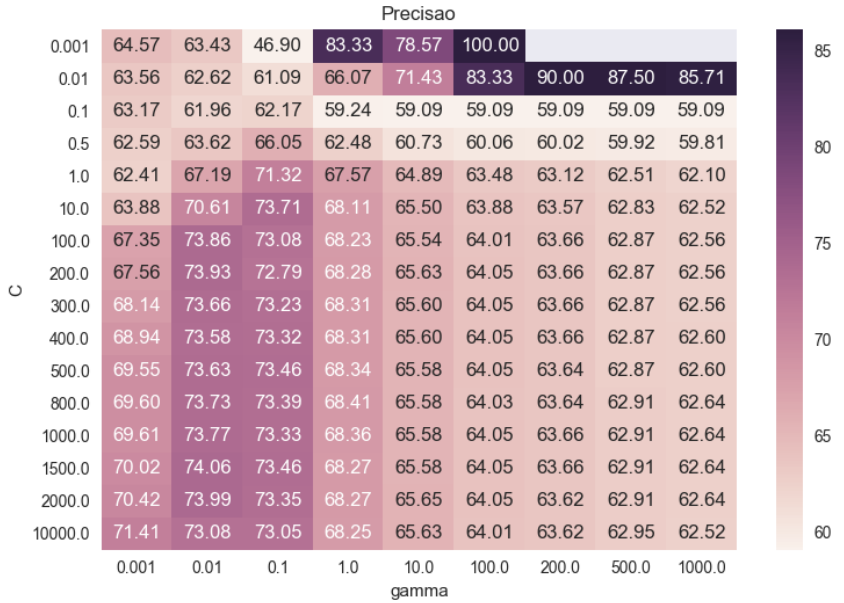
```
sklearn.metrics.precision_score()
sklearn.metrics.recall_score()
```

Resultados

Otimização dos parâmetros

Parâmetros da função de otimização:

- C (custo dos parâmetros)
- gamma (dispersão do kernel)



Avaliação do modelo

	Acurácia	Precisão	Recall
Machine Learning	74,3%	73,5%	88,4%
Horch*	48,2%	61,8%	32,1%
Yamashita*	54,7%	59,5%	73,3%
Singhal*	47,3%	59,1%	35,0%
Hagglund*	55,6%	58,2%	87,7%

* Métodos implementados por J. Dambros (UFRGS/DEQUI/GIMSCOP). Ver [Dambros, 2016]

Conclusões e trabalhos futuros

Conclusões e trabalhos futuros

Conclusões

- Machine Learning mostra-se muito interessante para este tipo de detecção de padrão
- Python como ferramenta para aplicação de ML torna o processo muito fácil
- Python + Machine Learning pode “democratizar” a extração de conhecimento a partir apenas de dados

Conclusões e trabalhos futuros

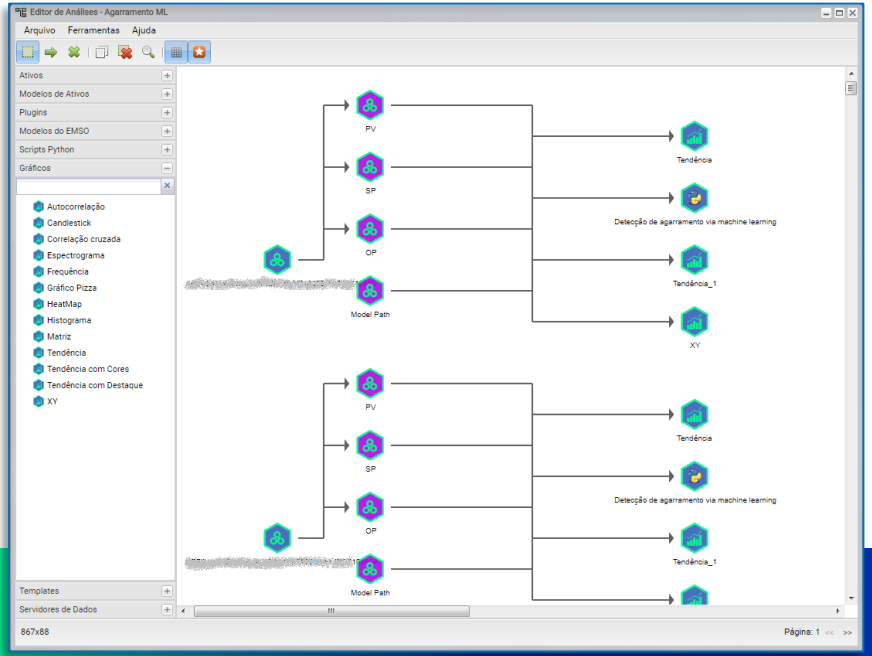
Trabalhos futuros

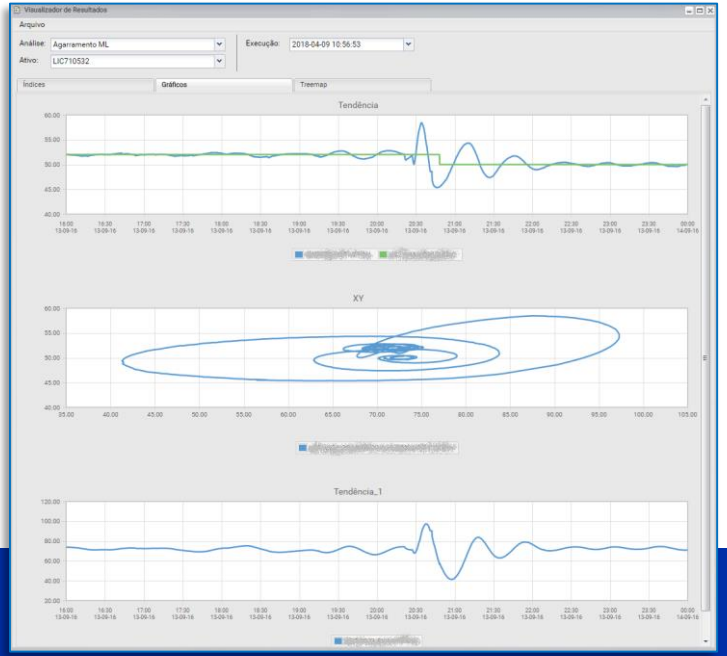
- Há ainda muito potencial de melhoria no modelo
  - Melhorias na simulação (válvula, controlador, processo, ruído branco, distúrbios)
  - Adição de mais dados reais
  - Testes mais extensivos de outros classificadores
  - Determinação de parâmetros ótimos do pipeline (ciclos, amostragem, clusters, ...)
  - Adição de outros atributos ao modelo (parâmetros ARMA, estatísticas descritivas, tipo da variável controlada, ...)
  - Análise detalhada dos casos de falso positivo

Conclusões e trabalhos futuros

Trabalhos futuros

- Pipeline de uso já implementado na forma de plug-in Python no PGA
- Testes mais amplos com dados aplicações industriais





Agradecimentos e bibliografia

Agradecimentos e bibliografia

Agradecimentos

- Elipse e Unisinos
- Equipe UFRGS/DEQUI/GIMSCOP
    - Jorge Otávio Trierweiler
    - Marcelo Farenzena
    - Jônathan Dambros
- Equipe Trisolutions
- Andrea Farias & Clara Kempf

Agradecimentos e bibliografia

Bibliografia

- [Bialkowski, 1992] – Bialkowski, W. L. (1992). “Dreams vs. Reality: A view from both sides of the gap”. Control systems (pp. 283-294). Whistler, BC, Canada.
- [Dambros, 2016] – Dambros, J. W. V. (2016). “Detecção do agarramento em válvulas de controle para sinais com referência variável”. Dissertação de mestrado, UFRGS/DEQUI/GIMSCOP, Porto Alegre, Brasil.
- [Horch, 1999] – Horch, A. (1999). “A simple method for detection of stiction in control valves”. Control Engineering Practice, v. 7, n. 10, pp. 1221-1231.
- [Yamashita, 2006] – Yamashita, y. (2006). “An automatic method for detection of valve stiction in process control loops”. Control Engineering Practice, v. 14, n. 5, pp. 503-510.
- [Singhal, 2005] – Singhal, A.; Salsbury, T. (2005). “A simple method for detecting valve stiction in oscillating control loops”. Journal of Process Control, v. 15, n. 4, pp. 371-382.
- [He, 2007] – He, Q. P. et al. (2007). “A curve fitting method for detecting valve stiction in oscillating control loops”. Industrial and Engineering Chemistry Research, v. 46, n. 13, pp. 4549-4560.

OBRIGADO!

ARIEL@TRISOLUTIONS.COM.BR

Agradecimentos e bibliografia

Bibliografia

- [Venceslau, Guedes, Silva, 2012] – Venceslau, A. R. S. et al. (2012). “Artificial neural network approach for detection and diagnosis of valve stiction”, Emerging Technologies & Factory Automation (ETFA), IEEE 17th Conference, Krakow, Poland.
- [Choudhury, 2008] – Choudhury, S. A. M. A. et al. (2008). “Diagnosis of Process Nonlinearities and Valve Stiction: Data Driven Approaches”. Springer - Berlin, Heidelberg
- [Jelali e Huang, 2010] – Jelali, M.; Huang, B. (2010). “Detection and Diagnosis of Stiction in Control Loops: State of the Art and Advanced Methods”, Springer – London, Dordrecht, Heidelberg, New York.
- [Choudhury, 2005] – Choudhury, M. A. A. S. et al. (2005). “Modelling valve stiction”. Control Engineering Practice, v. 13, pp. 641-658
- Ng, A. “Aprendizagem Automática”. Curso à distância, Universidade de Stanford. https://www.coursera.org/learn/machine-learning