FLOWCAST

Explainable Machine Learning

SFPython

2019, March 13

Outline

FLOWCAST

1. Machine learning intro
2. Why explanations are important
3. How to produce explanations currently in Python
    a. Global
    b. Local
4. How to make them better

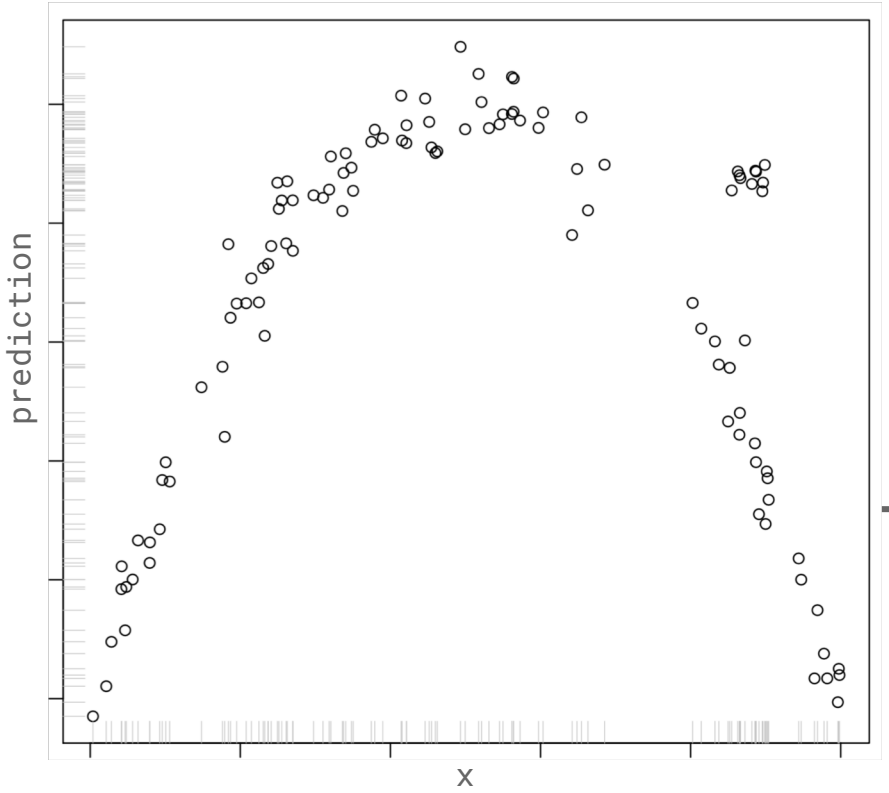
Machine Learning Intro

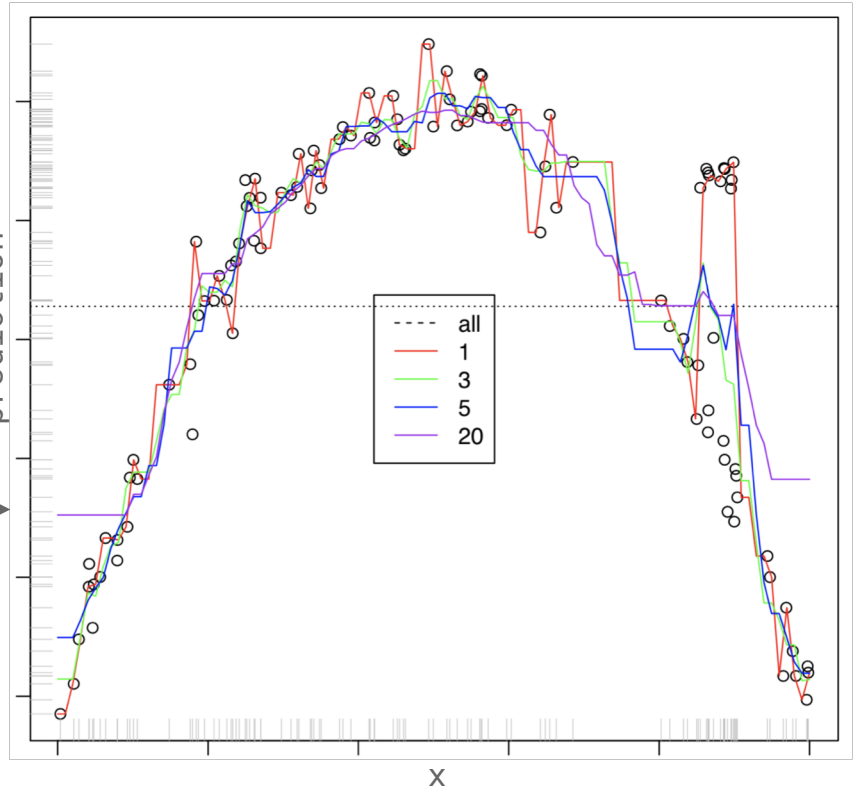
FLOWCAST

1. Machine learning approximates the world through evidence
2. At low dimensionality, it looks like simple curve fitting
3. Good generalizability is helpful to predict areas where data is sparse.
4. Overfitting is when the model fits to noise.
5. Hyperparameters are parameters of the model
6. Normally there are many inputs (features / dimensions) to a model (100s, 1000s, 10e4+)

Machine Learning Intro

FLOWCAST





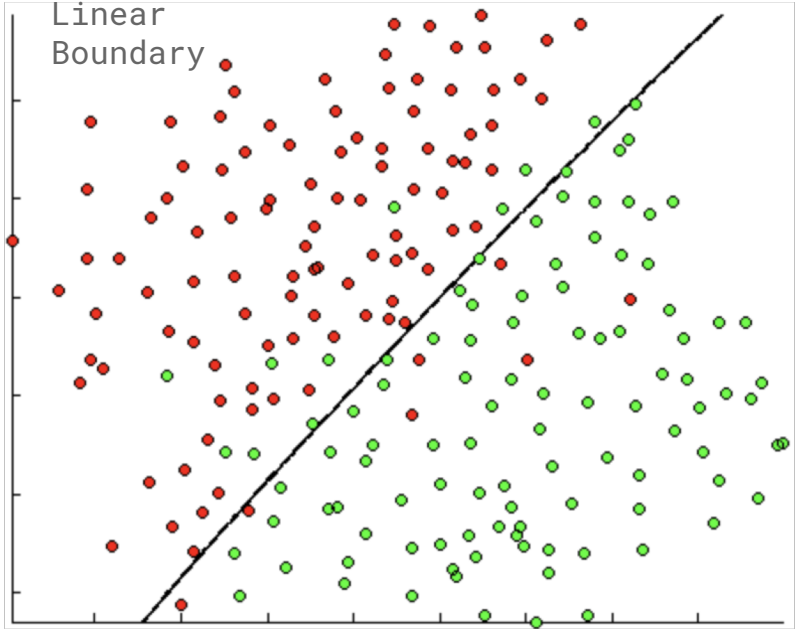
Cosma Rohilla Shalizi, **Advanced Data Analysis from an Elementary Point of View,** 2016 edition, pg 31 and 42

Machine Learning Classification

FLOWCAST

- ML classification aims to segregate classes (decrease entropy)
- Data Scientists' aim with ML is to improve predictive power (usually)
- Linear models are the least complex ML model (besides a simple average)

  complexity = (1 coefficient per dimension) + 1

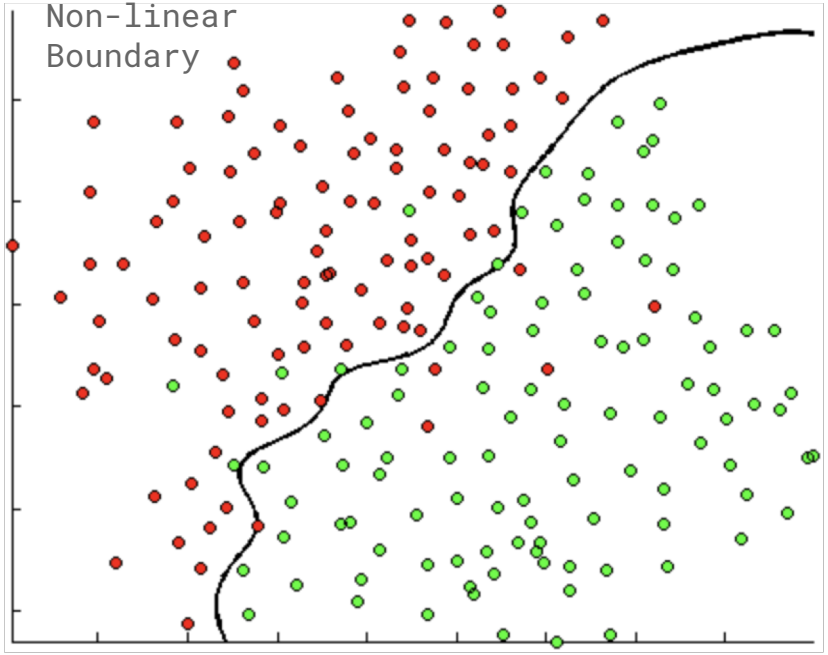


Andrew Ng, **OpenClassroom Machine Learning**, Exercise 8, http://openclassroom.stanford.edu

Machine Learning Classification

FLOWCAST

- Non-linear models improve predictive power at the cost of increased complexity
- Non-linear models are called “black-box models” in laymen’s
- complexity = (>>1 coefficient per dimension) + 1

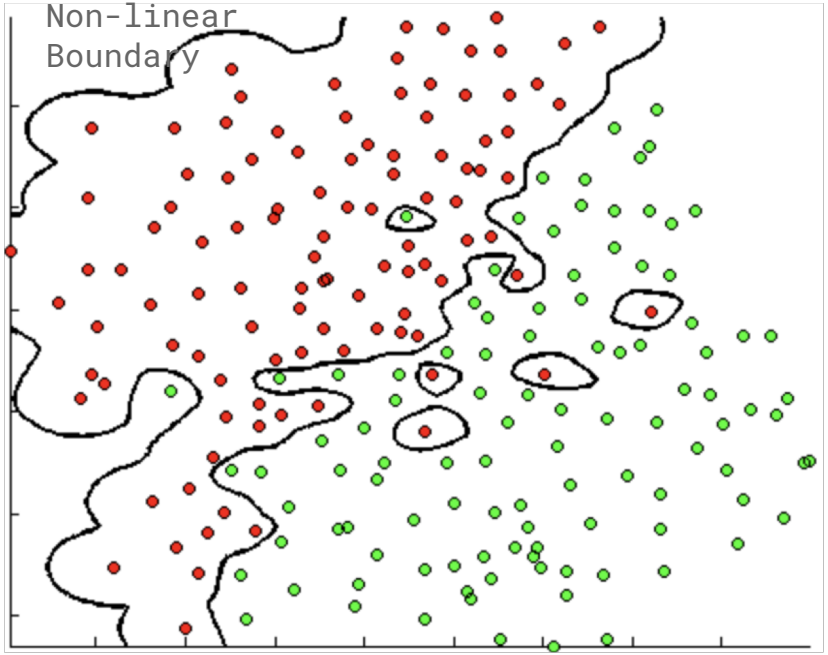


Andrew Ng, **OpenClassroom Machine Learning,** Exercise 8, http://openclassroom.stanford.edu

Machine Learning Classification

FLOWCAST

- But can lead to overfitting, decreasing the predictive power.
- On the training data, the cases are perfectly segregated. Will this generalize?



Andrew Ng, **OpenClassroom Machine Learning**, Exercise 8, http://openclassroom.stanford.edu

Machine Learning

FLOWCAST

Structured, numeric, non-null data matrix

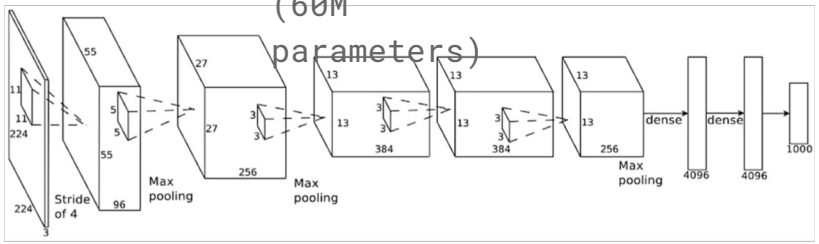
- _____
- _____
- _____
- _____
- _____

ML Algorithm

Set of predictions

256 x 256 x 3 (rgb) = 196,608 input params

AlexNet (2012) (60M parameters)



95% “greatest boy ever”

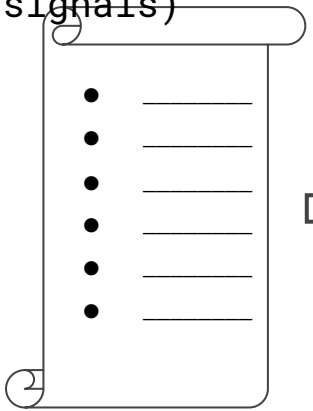
5% “silliness”

A Krizhevsky, I Sutskever, GE Hinton - Advances in neural information processing systems, 2012

Why are explanations important?

FLOWCAST

Input data (current vital signals)

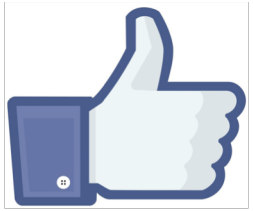


Robot Doctor

ML Model

You're going to die in the next 24 hours

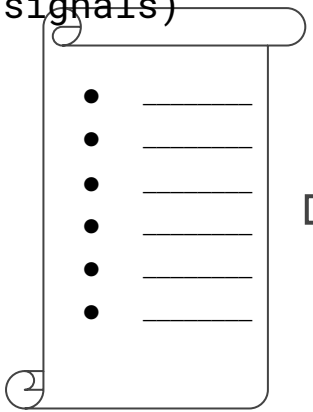
Great!



Why are explanations important?

FLOWCAST

Input data (current vital signals)



Robot

ML Model

You're going to die in the next 24 hours

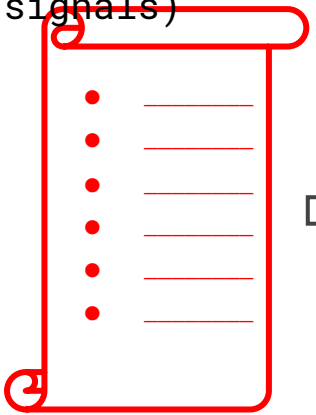
The top factors for the prediction are:

1. Heart rate = 140 bpm
2. Blood pressure = 170 over 110
3. Lower abdomen pain is True

Why are explanations important?

FLOWCAST

Input data (current vital signals)



Robot

ML Model

You're going to die in the next 24 hours

The top factors for the prediction are:

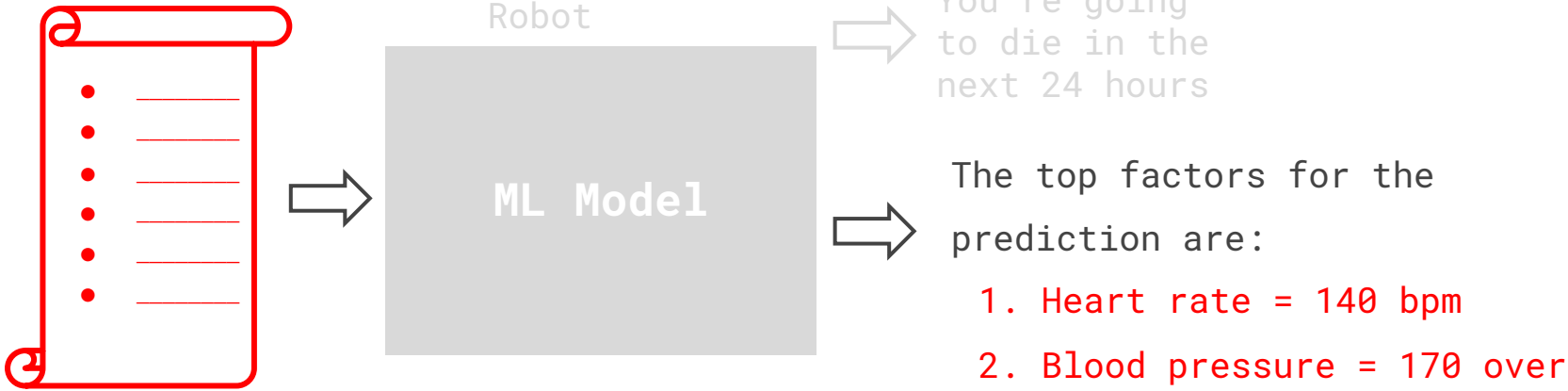
1. Heart rate = 140 bpm
2. Blood pressure = 170 over 110
3. Lower abdomen pain is True

Why are explanations important?

FLOWCAST

Our Explanation Requirements:

1. Reasons come from the inputs
2. Prediction is fixed
3. Model is fixed



Global Explanations

FLOWCAST

Pros:

- Fast, easy way to discover which features were impactful
- Feature importance works with any tree-based ensemble model

Cons:

- Not local

```
import pandas as pd

def model_importances(rf_model, columns):
    """
    Return a dataframe of features and the model importances from the randomforest
    """
    return (pd.DataFrame({'features' : columns,
                          'importances' : rf_model.feature_importances_})
            .sort_values('importances', ascending = False))
```

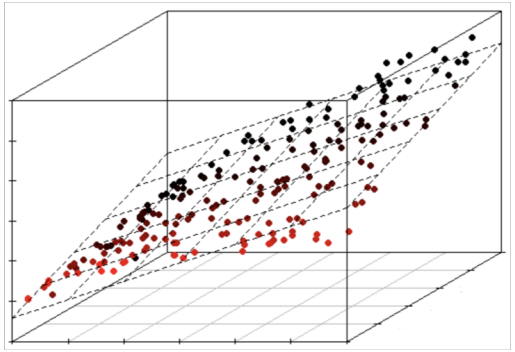
	importance
duration	0.285520
balance	0.094603
age	0.091508
day	0.080745
success	0.053823
pdays	0.039141

Local Explanations

FLOWCAST

Linear Model

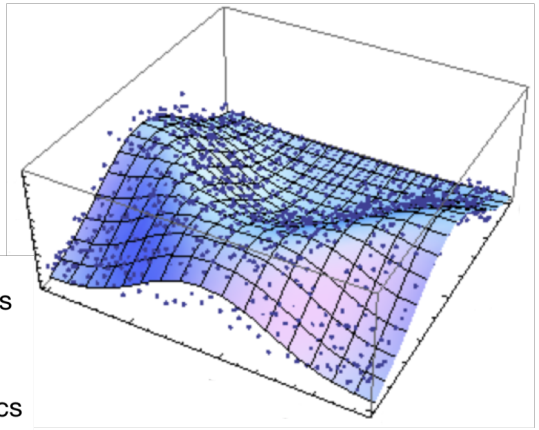
$$f(x) = x_o + \sum_{i}^{ndim} C_i x_i$$



Nonlinear Model

$$\mathcal{F}(x) = x_o + \sum_{i}^{ndim} C_i \cdot g_i(y)$$

$$g(x) = \begin{cases} \sum_n a_n \cdot e^{i\frac{2\pi n x}{P}} & \text{Fourier Series} \\ \sum_i a_i \cdot (x - b_i)^i & \text{Taylor Series} \\ \dots & \text{Other generics} \end{cases}$$



- Feature impact is coefficient size (assuming normalized x_i)
- Explanations at each location in phase space is the same
- Doesn't capture dynamics of the data

- Feature impact is the gradient of the model at the prediction
- Explanations are different in different areas of feature phase space

Local Explanations

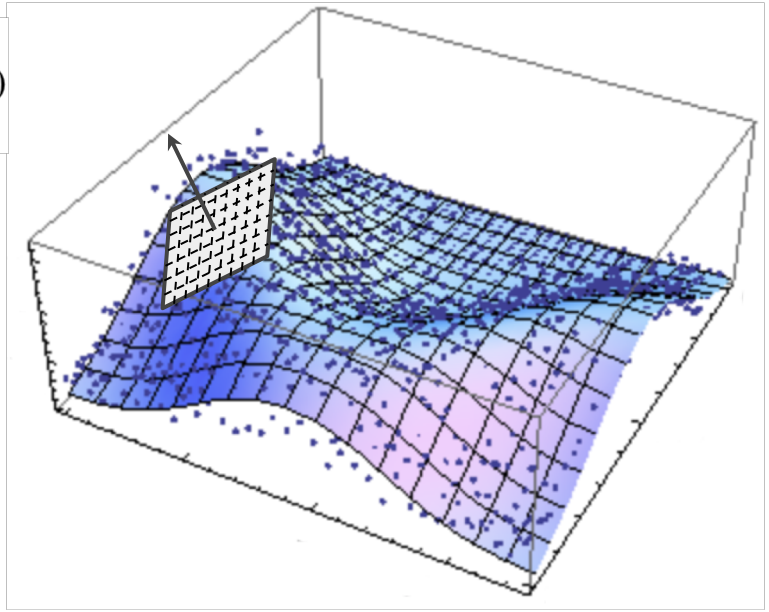
FLOWCAST

- Feature impact is the gradient of the model at the prediction
- To first approximation, fit a linear model local to the prediction. This is what LIME does

$$\mathcal{F}(x) = x_o + \sum_{i}^{ndim} C_i \cdot g_i(y)$$

$$\overrightarrow{\text{imp}}(x) = \nabla \cdot \mathcal{F}(x)$$

Nonlinear Model



LIME

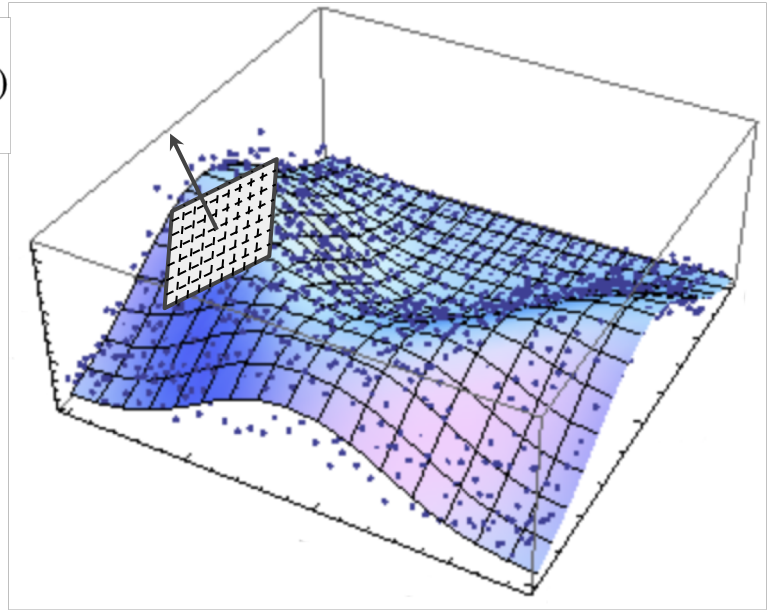
FLOWCAST

- LIME is “Locally Interpretable Model Estimation”, or sampling data near a prediction and fitting a local linear model
- It has a Python package!

$$\mathcal{F}(x) = x_o + \sum_{i}^{ndim} C_i \cdot g_i(y)$$

$$\overrightarrow{\text{imp}}(x) = \nabla \cdot \mathcal{F}(x)$$

Nonlinear Model



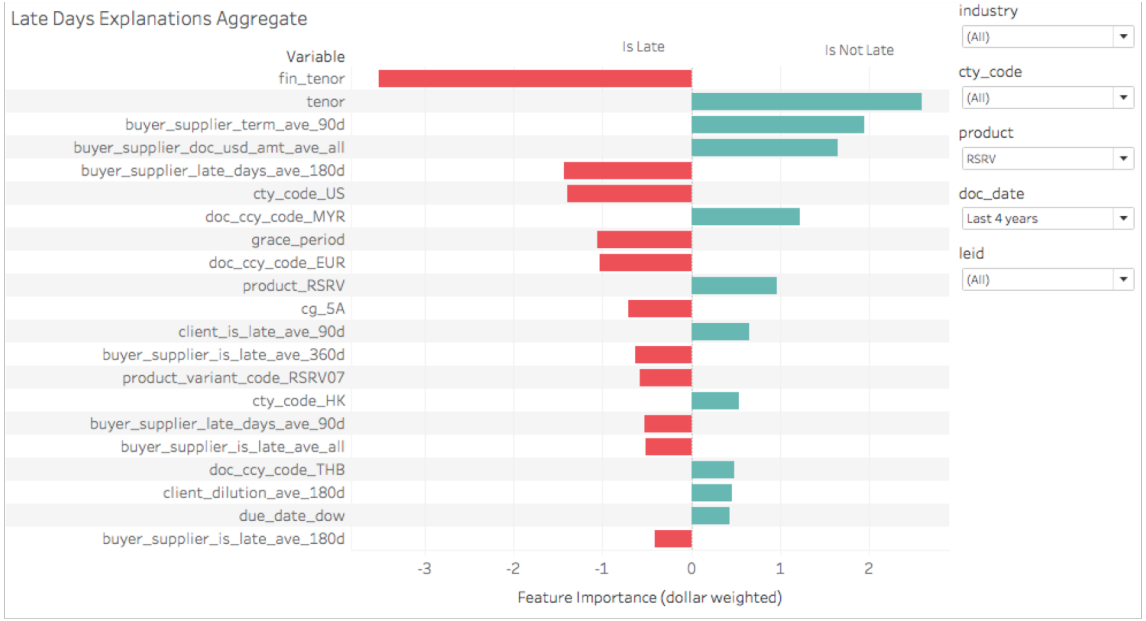
LIME Demo

https://marcotcr.github.io/lime/tutorials/

Beyond LIME

FLOWCAST

- In practice, LIME output is hard to understand and read, even for professionals in the field
- Correlated fields will bunch up in the explanations
- How can we make this more intuitive?



Beyond LIME

FLOWCAST

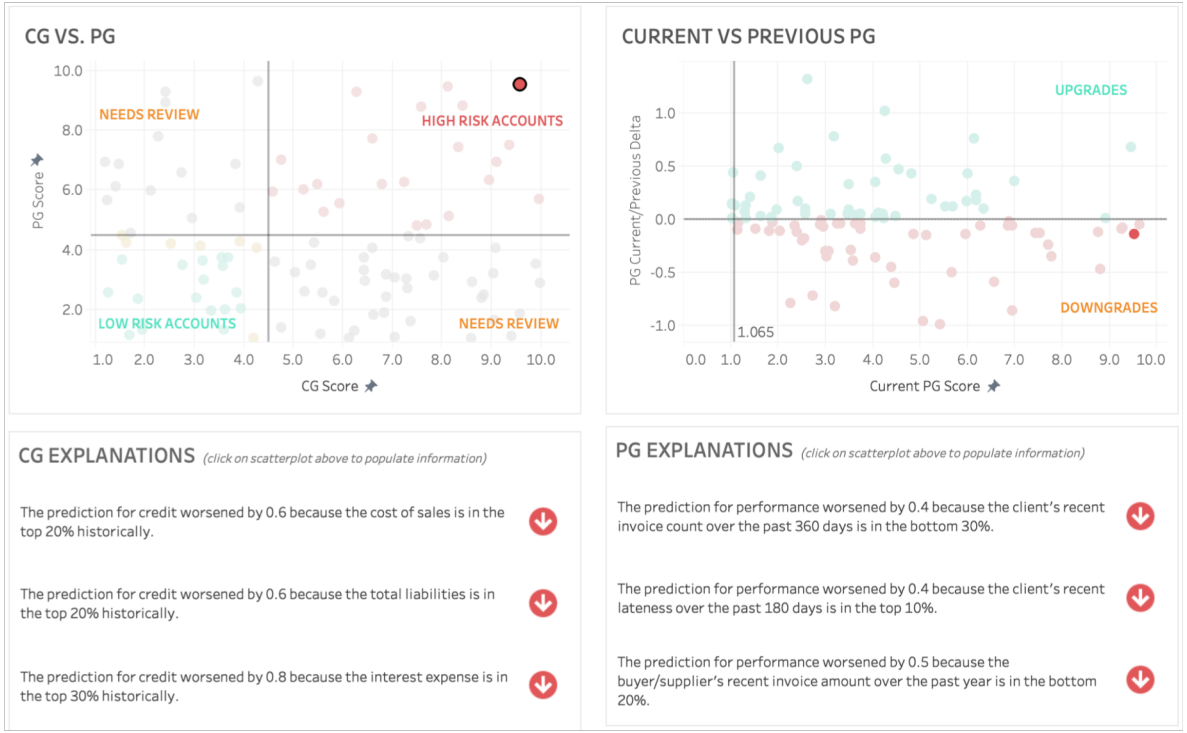
1) Group correlated (similar) features
   - Compute correlation matrix
   - Group features above a threshold abs(> 0.9)
   - Take cumulative feature impact from LIME

Beyond LIME

FLOWCAST

2) Comparison to peer group
   - Take the top N features output from LIME and compare the value vs. peer group.

3) Auto-generate sentences



Thanks!

FLOWCAST

We’re hiring!

- Python Software Engineer
- Data Scientist
- Data Engineer
- Machine Learning Engineer

Please track me down after! Or email directly:

eitan@flowcast.ai