Model Evaluation For Humans

About me

Husband and Father
Philly data scientist
Wrote a haiku once

https://dantegates.github.io/2019/01/07/model-evaluation-for-humans.html
(https://dantegates.github.io/2019/01/07/model-evaluation-for-humans.html)

Disclaimers

- most useful for new(er) data scientists
- but there may be take aways for everyone
- oversimplication
- guidelines/examples are not exhaustive
- breadth not depth
- ask me how I know...

Goal of this talk

Discuss guidelines and practices for model evaluation

Motivation: all models are wrong

"All models are wrong, ..."

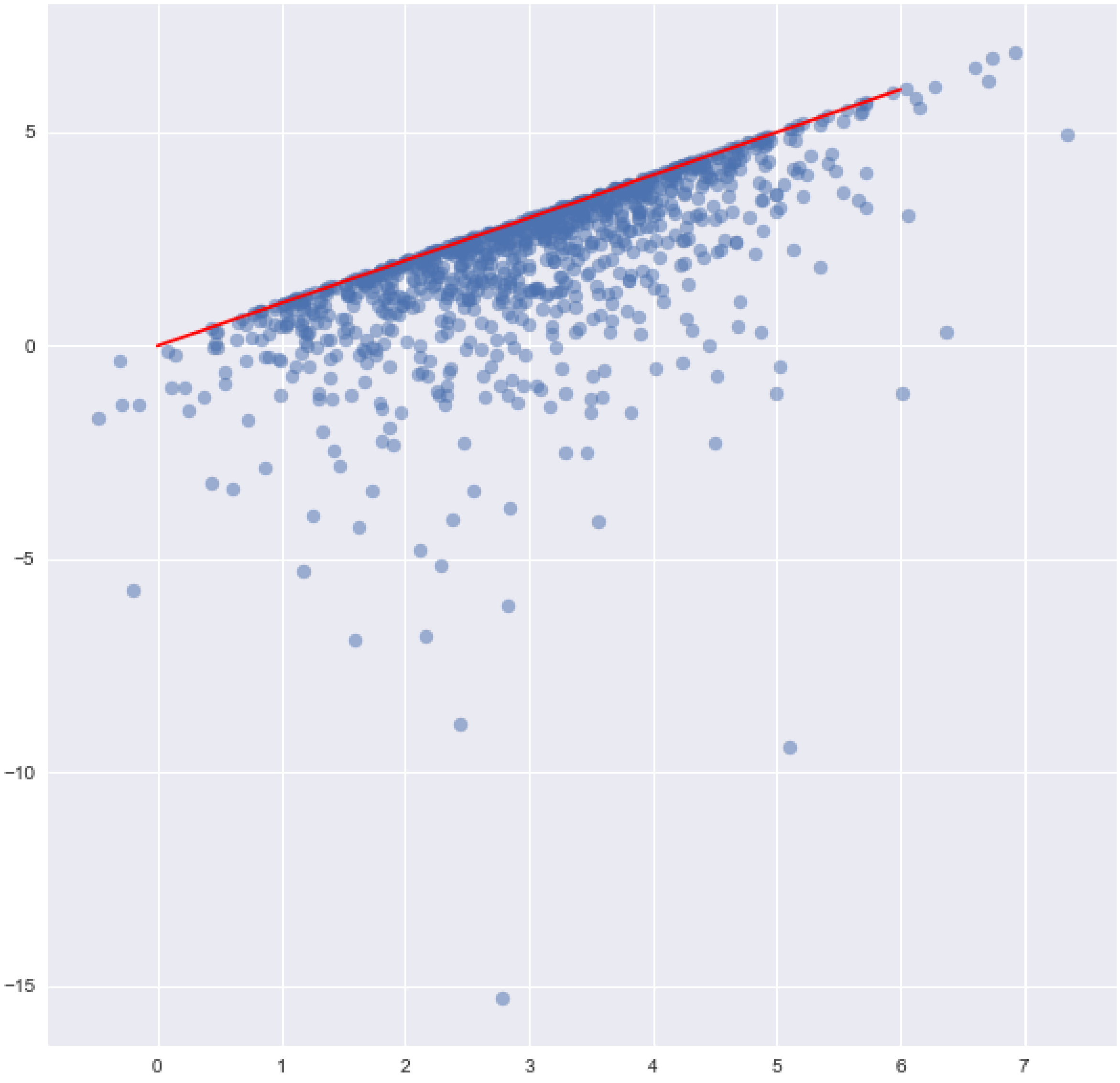
$$\underset{\theta}{\text{argmin}}\ L(y, \hat{y})$$

$$MLE := \underset{\theta}{\text{argmax}}\ \pi(D \mid \theta)$$

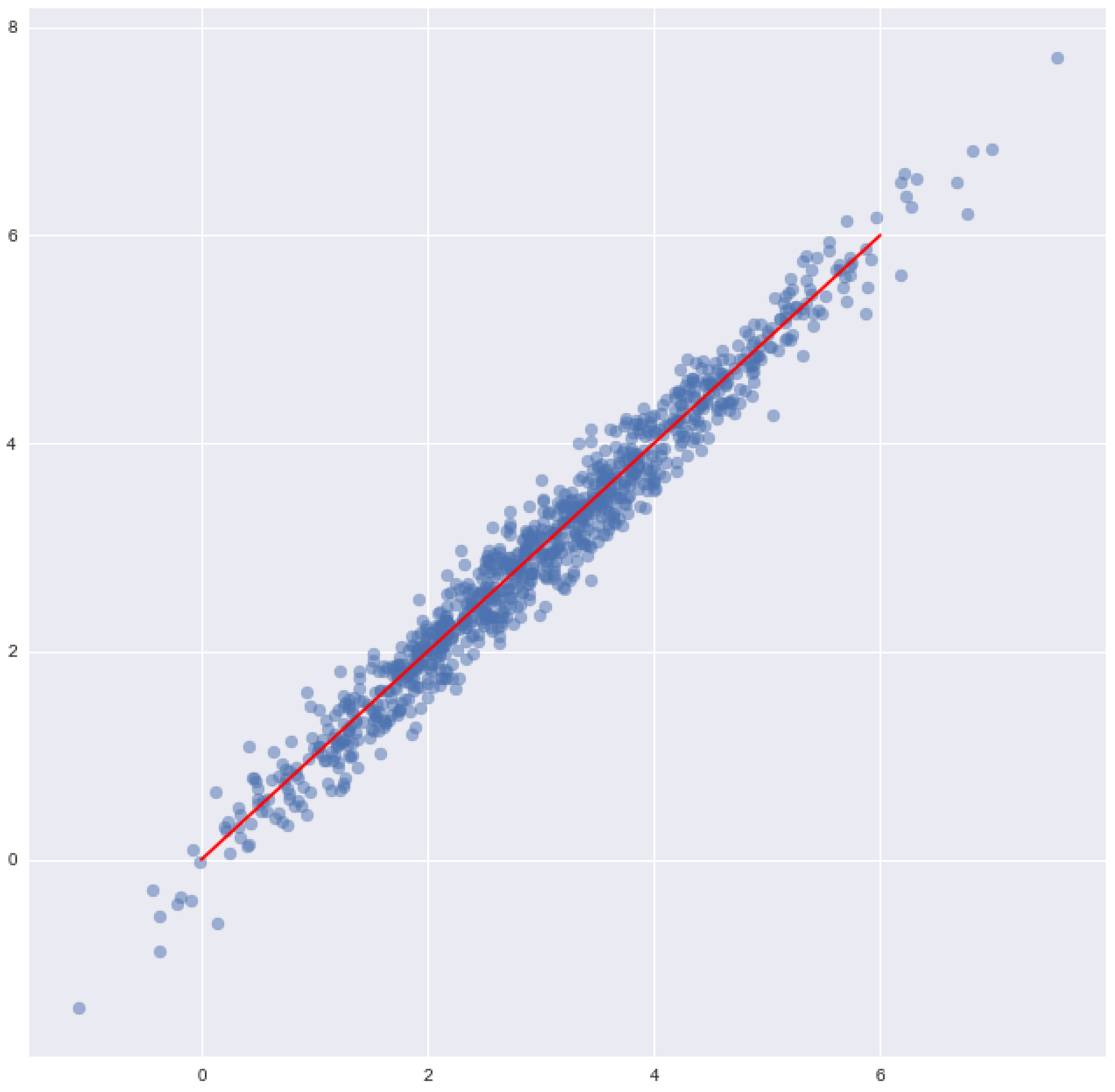
$$\hat{y} = f(x) + \epsilon$$

$$P(\theta \mid D) \propto P(D \mid \theta)P(\theta)$$

Ask me how I know…


5
0
−5
−10
−15
0
1
2
3
4
5
6
7

On a better day... still wrong


8
6
4
2
0
0
2
4
6

The zen of model evaluation

"*All models are wrong, some are useful*"

The zen of model evaluation summarized

- All models are wrong
- Some are useful
- Others are not
- Usefulness counts

Model evaluation and the product lifecycle

- Before development
  - Business & Data Understanding
- During development
  - Data Preparation → Modeling → Evaluation
- Deployment and beyond
  - Deployment → Evaluation & Monitoring → Iteration

Before development

- Highly context specific
- Essential for rest of project

Hazards of getting this wrong

- Model does not solve POI
- Model indirectly solves POI
- Model cannot be evaluated in prod
    - Reduced confidence in model
- Diminishing returns
- Iteration is impaired

Some things to think about...

Think about framing the problem

- How will this model be evaluated?
- How will success be *measured*?
- Choose metrics and KPIs
    - Choose metrics carefully (more on this later, ... maybe)

Think about deployment and future iterations

- Prepare for deployment
- Prepare for future iterations
- Nothing is more permanent than temporary
- Example: A/B testing

Think about the feedback loop

- How will outcomes from the model be collected?
- Example: Filtered predictions
- Example: Ad recommendations
- Example: Intervention for hospital readmission

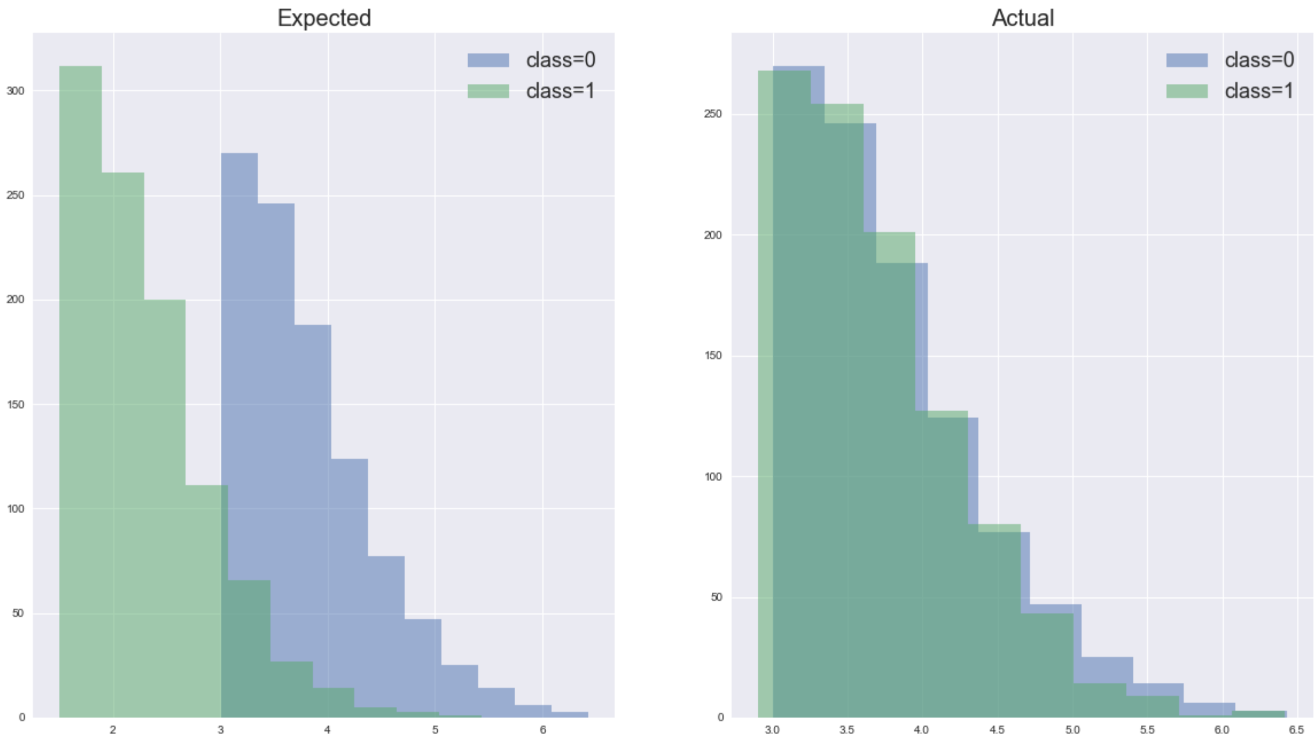
During development

- Most generalizable
- Opportunity adopt and standardize best practices

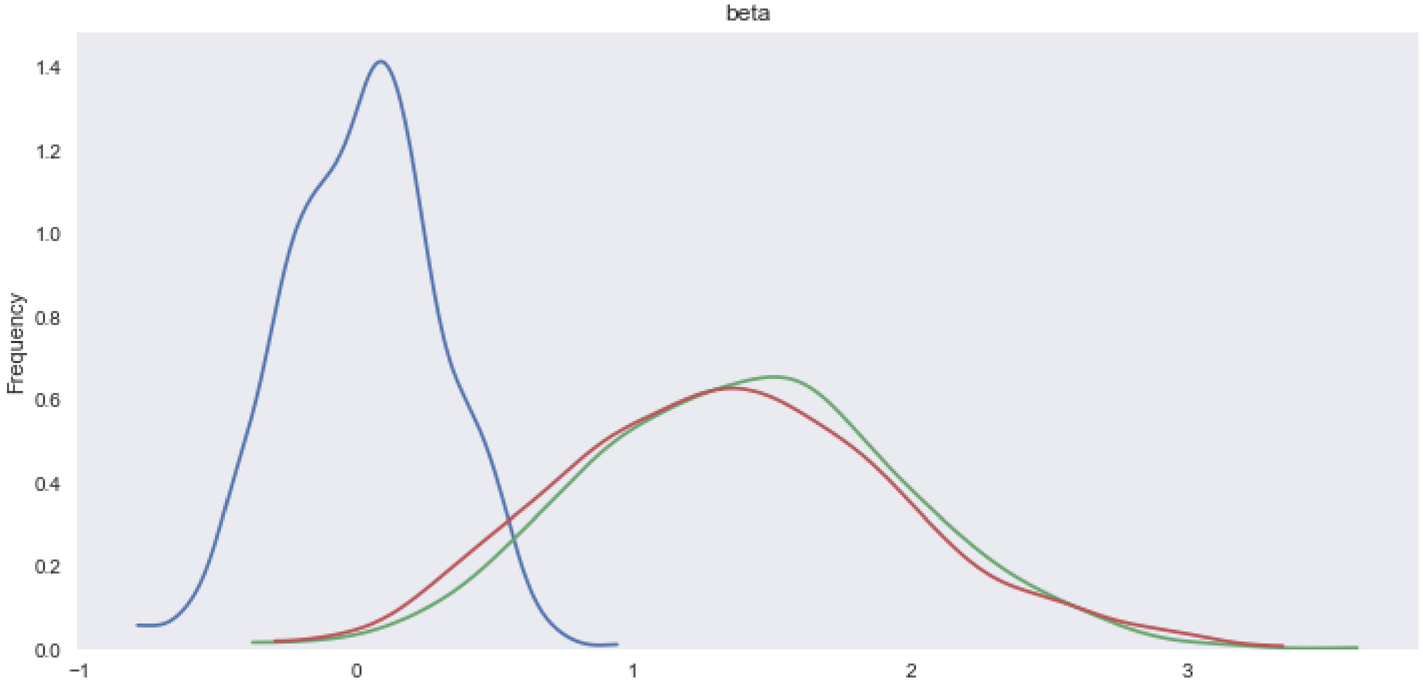
Test assumptions

- Check distributions of data
- Plot, plot, plot
- Check features against conventional wisdom
- Etc.
- $f(\quad) =$

Testing assumptions: Example



Test assumptions: Example



Establish a baseline

A good baseline is

- Easy to implement
- Easy to understand

Examples of baselines

- A simple heuristic
  - Example: Ratings
  - Example: Persistency
- An interpretable model
  - Example: Comorbidities
- The existing model or methodology
- Acceptance criteria

Sanity Checks

- Inspecting learned parameters
  - Example: Comorbidities

- Examining outliers in the model error

- Algorithm specific assumptions

- Examine range of predictions

Cross validation

Replicate the deployment pattern

- Avoid leakage from future examples
- More like backtesting, less like K-fold CV

```
months = list(df.groupby('month'))
for (m1, df1), (m2, df2) in zip(months, months[1:]):
    X_train, y_train = df1[features], df1[response]
    X_test, y_test = df2[features], df2[response]
```

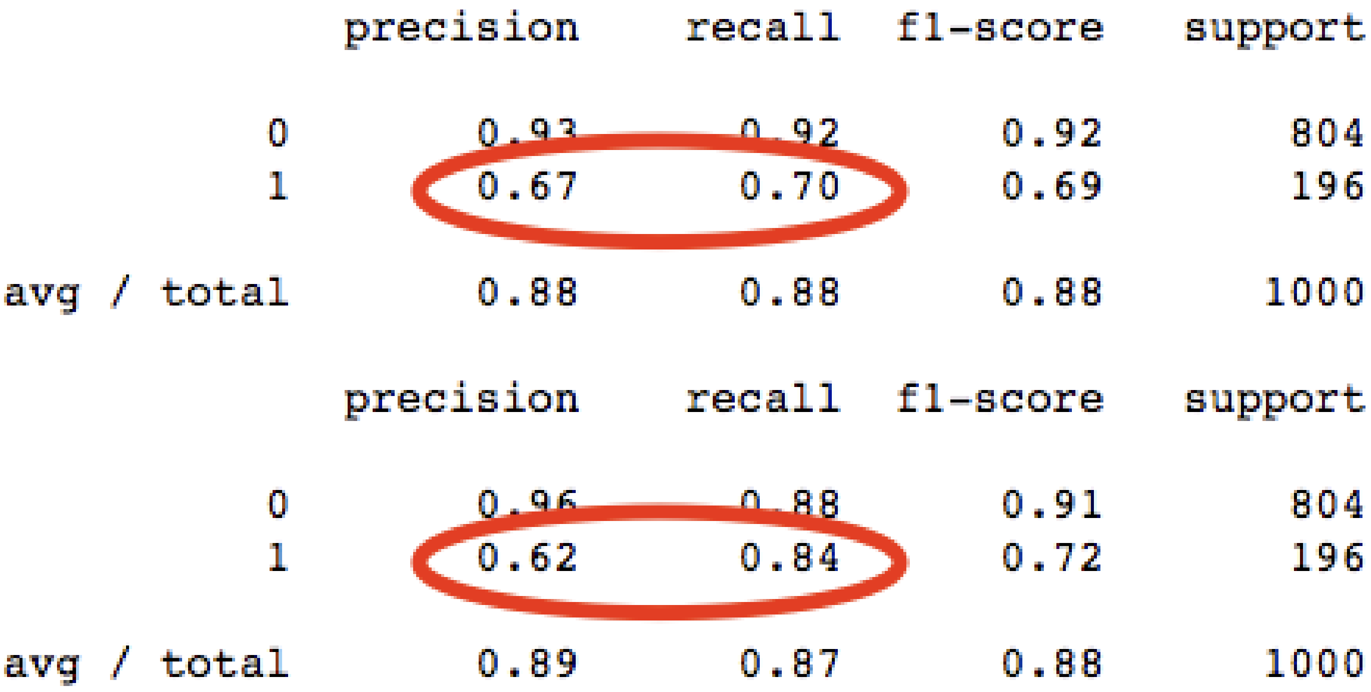
Train on a cleaner data set

- Validate modeling approach apart from noise in data set
  - Example: Ratings

Deployment and beyond

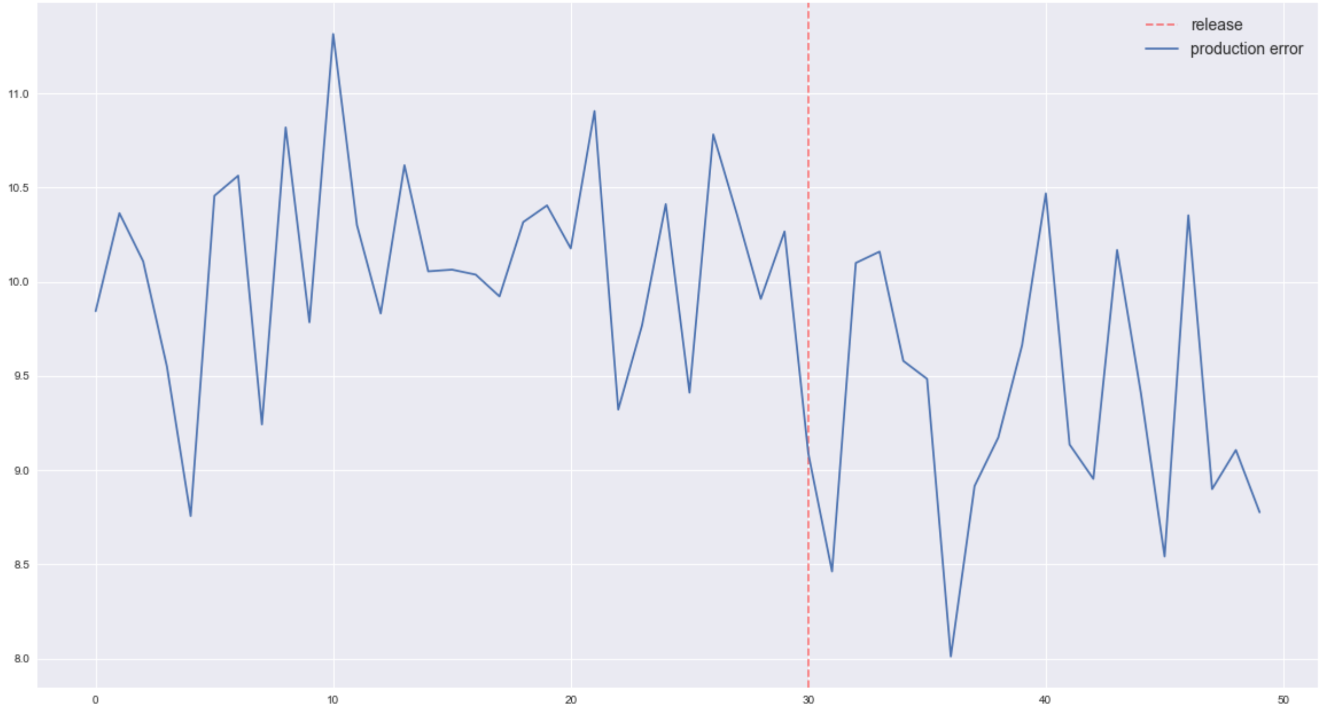
- This is what really matters (Remember the Zen!)
- Ensure that your model is (and stays) useful!
- Iterate with confidence!

A/B testing



```
             precision    recall  f1-score   support

          0       0.93      0.92      0.92       804
          1       0.67      0.70      0.69       196

avg / total       0.88      0.88      0.88      1000
             precision    recall  f1-score   support

          0       0.96      0.88      0.91       804
          1       0.62      0.84      0.72       196

avg / total       0.89      0.87      0.88      1000
```

A/B testing



release
production error
11.0
10.5
10.0
9.5
9.0
8.5
8.0
0
10
20
30
40
50

Monitoring: Covariate shift

- Check for deviations in run time data
- Keep it simple
- Keep it actionable

Monitoring: Sanity checks

- Remember me?
- Use into QA
- Integrate in automatic training pipelines

A word on metrics

- Conversion to business value
- Comparability
  - Across time, data, iterations, etc.

Conclusion

- Hopefully you have some take aways
- Share your suggestions!

Questions?