CSCI 567: Machine Learning Discussion – Evaluation Metrics

Xiao Fu

April 2024

Materials from: *https://developers.google.com/machine-learning/crash-course*

Credit to Wang (Bill) Zhu - TA 2023 Fall

Discussion Overview

- Cross Validation
- Evaluation Metrics

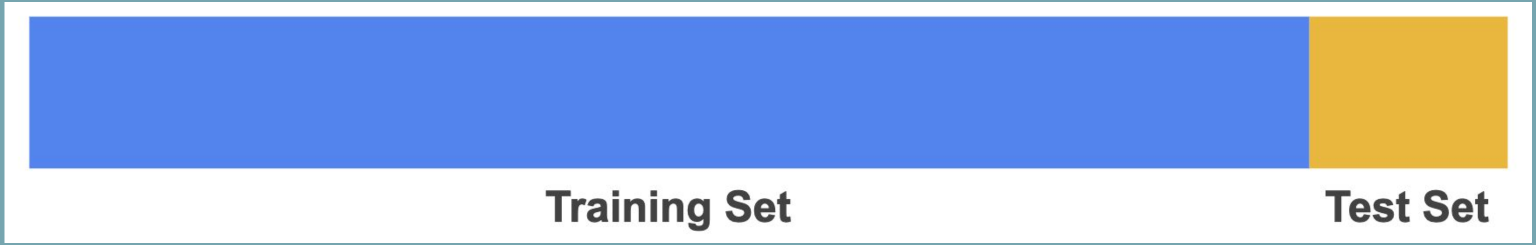
Cross-validation Overview

- Training and Test Sets
- Validation Set
- Cross-validation

Training and Test Sets

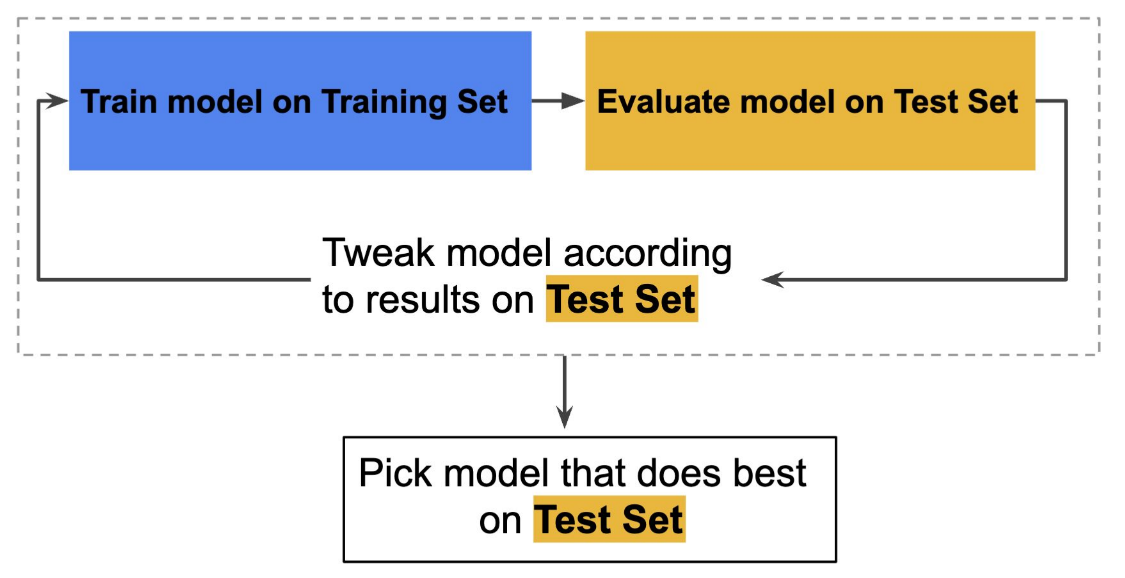
- Training set - a subset to train a model
- Test set – a subset to test a trained model

You could imagine slicing the single data set as follows (80%/20%):



Training and Test Sets

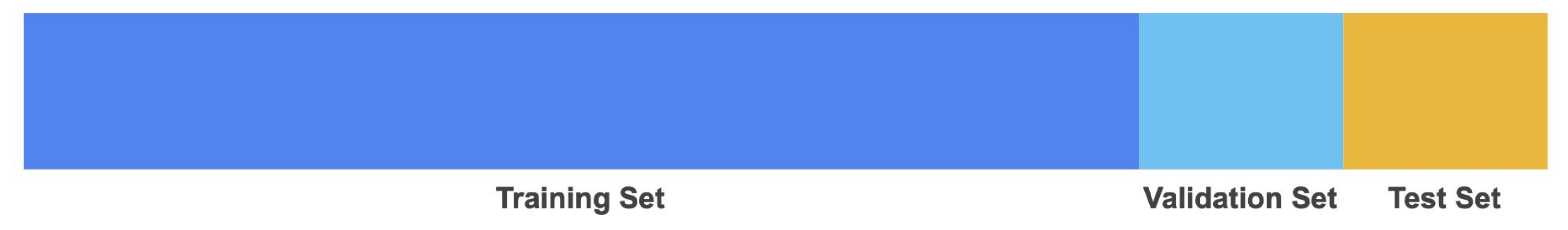
- With two partitions, the workflow is follows
- Overfitting problem



Train model on Training Set
Evaluate model on Test Set
Tweak model according to results on Test Set
Pick model that does best on Test Set

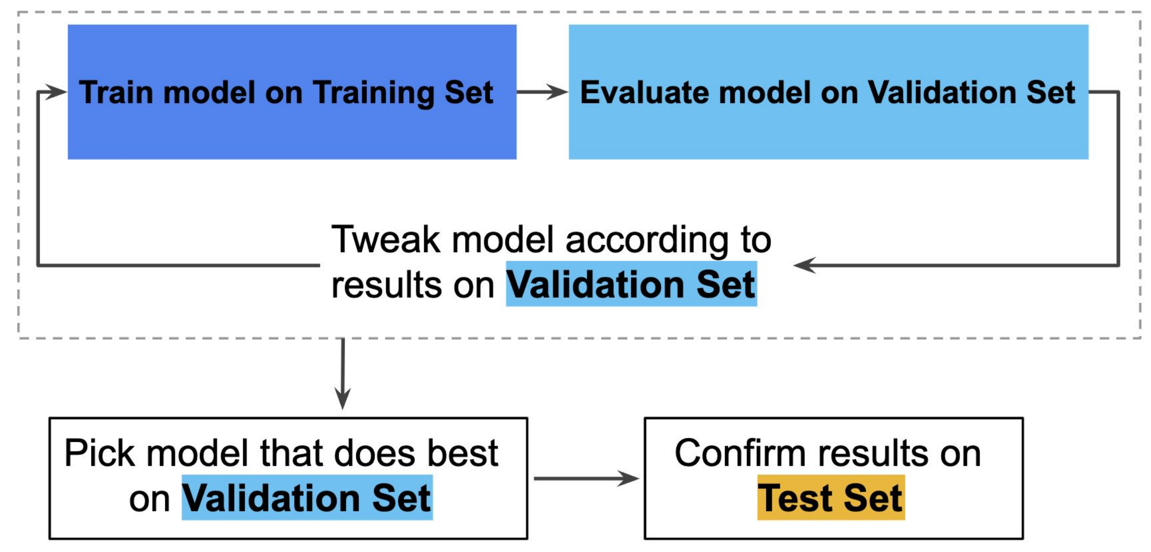
Validation Set

- Reduce your chances of overfitting
- Three subsets partition shown in the following figure



Validation Set

- Tune hyper-parameters (batch size, learning rate, etc.) on the validation set
- Then, use the test set to double-check your evaluation after the model has "passed" the validation set. (exam analogy: Lectures, HWs, Finals)



Train model on Training Set

Evaluate model on Validation Set

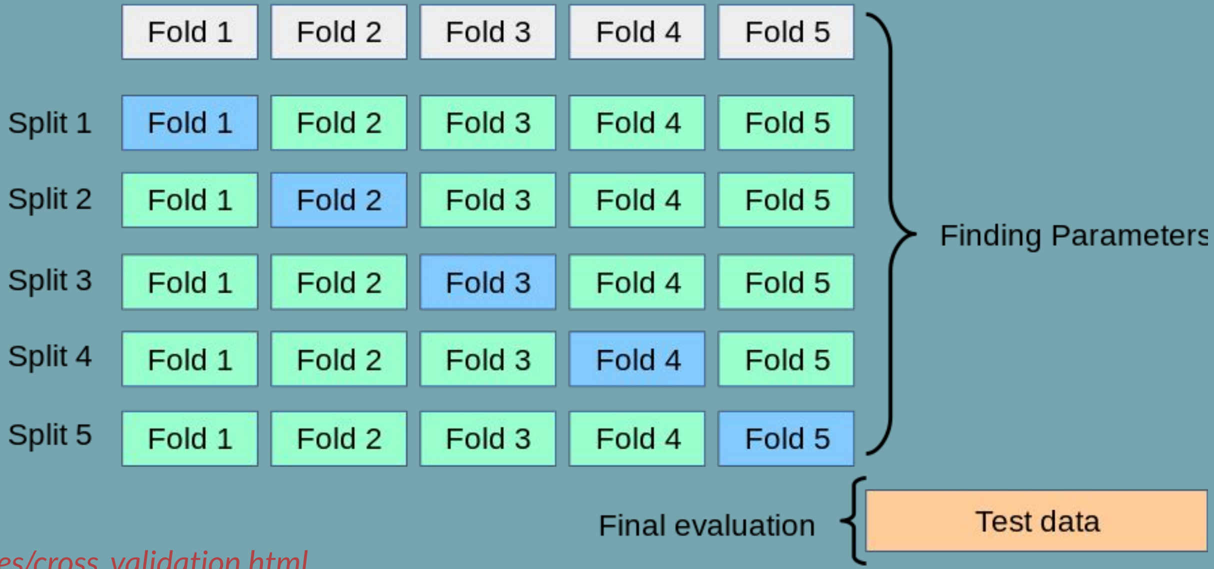
Tweak model according to results on **Validation Set**

Pick model that does best on **Validation Set**

Confirm results on **Test Set**

Cross-Validation: k-fold

- You need the validation set to be large (avoid overfitting)
- You need the validation set to be small (to have enough training data)
- Data set is limited



	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Split 1	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Split 2	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Split 3	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Split 4	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Split 5	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5

Image from: *https://scikit-learn.org/stable/modules/cross_validation.html*

Cross-Validation

- Split the data into k fold, use (k-1) fold for training and 1 fold for validation
- After finalizing hyper-parameters, use the entire training+validation to train the model

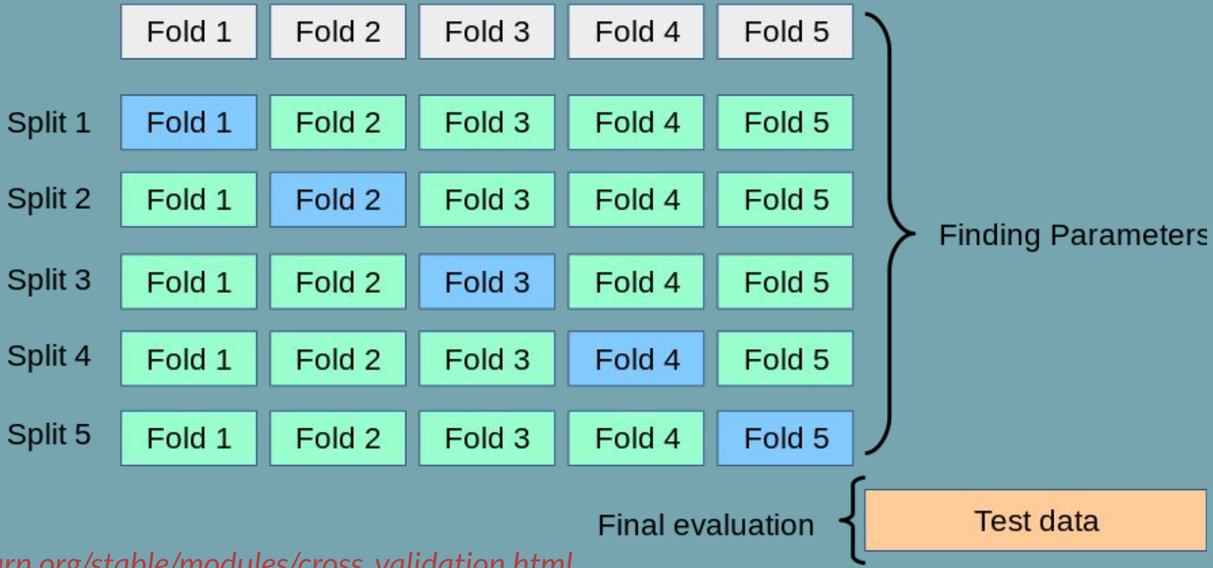


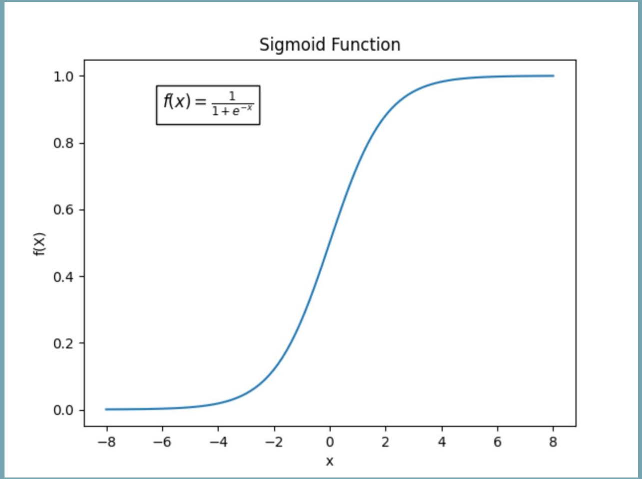
Image from: [*https://scikit-learn.org/stable/modules/cross_validation.html*](https://scikit-learn.org/stable/modules/cross_validation.html)

Evaluation Metrics Overview

- Thresholding
- Confusion matrix
- Accuracy
- Precision and Recall
- ROC and AUC
- Calibration

Thresholding - Logistic Regression

- Binary classification: $y = f(x), y \in \{0, 1\}$
- A logistic regression model outputs a probability in $(0, 1)$
- Choose a threshold to convert it to a binary value
- 0.5 is not always the best
- *Why? Depends on the evaluation metrics.*



Confusion Matrix – Tumor Prediction

- Use 2x2 confusion matrix to separate out different kinds of errors
- **Class-imbalanced** setup: 9% of examined tumors are malignant, 91% benign

True Positives (TP) Reality: Malignant ML predicted: Malignant	**False Positives (FP)** Reality: Benign ML predicted: Malignant Type-1 Error
False Negatives (FN) Reality: Malignant ML predicted: Benign Type-2 Error	**True Negatives (TN)** Reality: Benign ML predicted: Benign

Evaluation Metrics: Accuracy

- Accuracy is the fraction of predictions our model got right
- Accuracy can Be Misleading

True Positives (TP) Reality: Malignant ML predicted: Malignant Number of TP results: 1	**False Positives (FP)** Reality: Benign ML predicted: Malignant Number of FP results: 1
False Negatives (FN) Reality: Malignant ML predicted: Benign Number of FN results: 8	**True Negatives (TN)** Reality: Benign ML predicted: Benign Number of TN results: 90

Evaluation Metrics: Accuracy - Can Be Misleading

- Accuracy is the fraction of predictions our model got right
- $\text{Accuracy} = \frac{TP+TN}{TP+FP+FN+TN}$

True Positives (TP) Reality: Malignant ML predicted: Malignant Number of TP results: 1	**False Positives (FP)** Reality: Benign ML predicted: Malignant Number of FP results: 1
False Negatives (FN) Reality: Malignant ML predicted: Benign Number of FN results: 8	**True Negatives (TN)** Reality: Benign ML predicted: Benign Number of TN results: 90

Evaluation Metrics: Accuracy - Can Be Misleading

- Accuracy is the fraction of predictions our model got right
- Accuracy $= \frac{TP+TN}{TP+FP+FN+TN}$
- How about a model that predicts negative all the time?

True Positives (TP) Reality: Malignant ML predicted: Malignant Number of TP results: 1	**False Positives (FP)** Reality: Benign ML predicted: Malignant Number of FP results: 1
False Negatives (FN) Reality: Malignant ML predicted: Benign Number of FN results: 8	**True Negatives (TN)** Reality: Benign ML predicted: Benign Number of TN results: 90

Exercise (2 mins)

In which of the following scenarios would suggest that the ML model is doing a good job?

A. A deadly, but curable, medical condition afflicts .01% of the population. An ML model uses symptoms as features and predicts this affliction with an accuracy of 99.99%.

B. An expensive robotic chicken crosses a very busy road a thousand times per day. An ML model evaluates traffic patterns and predicts when this chicken can safely cross the street with an accuracy of 99.99%.

C. In the game of roulette, a ball is dropped on a spinning wheel and eventually lands in one of 38 slots. Using visual features (the spin of the ball, the position of the wheel when the ball was dropped, the height of the ball over the wheel), an ML model can predict the slot that the ball will land in with an accuracy of 50%.

Evaluation Metrics: Precision and Recall

- What proportion of positive identifications was actually correct?
- Precision $= \frac{TP}{TP+FP}$

True Positives (TP) Reality: Malignant ML predicted: Malignant Number of TP results: 1	**False Positives (FP)** Reality: Benign ML predicted: Malignant Number of FP results: 1
False Negatives (FN) Reality: Malignant ML predicted: Benign Number of FN results: 8	**True Negatives (TN)** Reality: Benign ML predicted: Benign Number of TN results: 90

Evaluation Metrics: Precision and Recall

- What proportion of positive identifications was actually correct?
- Precision $= \frac{TP}{TP+FP}$
- 0.5

True Positives (TP) Reality: Malignant ML predicted: Malignant Number of TP results: 1	**False Positives (FP)** Reality: Benign ML predicted: Malignant Number of FP results: 1
False Negatives (FN) Reality: Malignant ML predicted: Benign Number of FN results: 8	**True Negatives (TN)** Reality: Benign ML predicted: Benign Number of TN results: 90

Evaluation Metrics: Precision and Recall

- What proportion of actual positives was identified correctly?
- Recall $= \frac{TP}{TP+FN}$

True Positives (TP) Reality: Malignant ML predicted: Malignant Number of TP results: 1	**False Positives (FP)** Reality: Benign ML predicted: Malignant Number of FP results: 1
False Negatives (FN) Reality: Malignant ML predicted: Benign Number of FN results: 8	**True Negatives (TN)** Reality: Benign ML predicted: Benign Number of TN results: 90

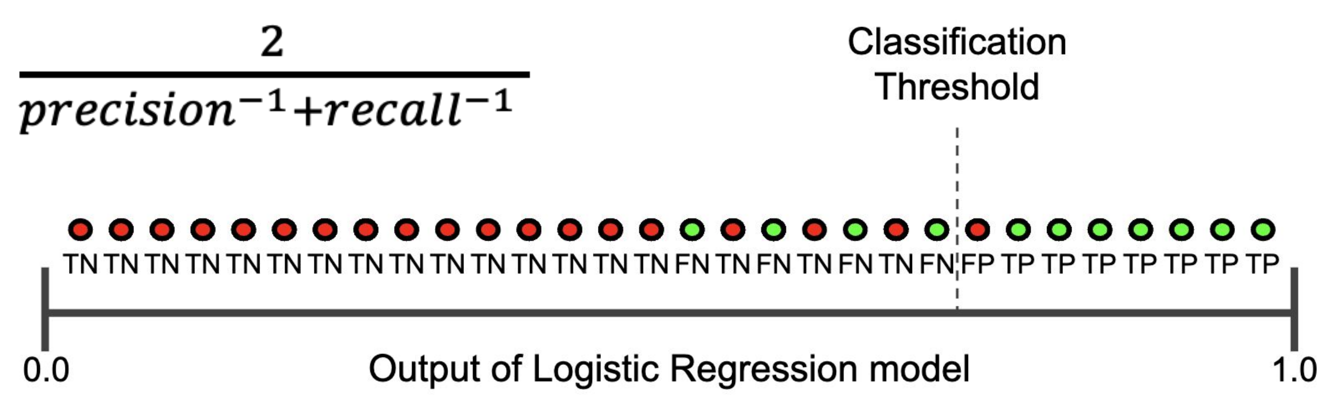
Evaluation Metrics: Precision and Recall

- What proportion of actual positives was identified correctly?
- Recall $= \frac{TP}{TP+FN}$
- 0.11

True Positives (TP) Reality: Malignant ML predicted: Malignant Number of TP results: 1	**False Positives (FP)** Reality: Benign ML predicted: Malignant Number of FP results: 1
False Negatives (FN) Reality: Malignant ML predicted: Benign Number of FN results: 8	**True Negatives (TN)** Reality: Benign ML predicted: Benign Number of TN results: 90

Precision and Recall: A Tug of War

- Hard to optimize both at the same time by changing threshold
- Precision $= \frac{TP}{TP+FP}$, Recall $= \frac{TP}{TP+FN}$
- F1 $= \frac{2}{precision^{-1}+recall^{-1}}$


Classification Threshold
TN TN TN TN TN TN TN TN TN TN TN TN TN TN TN FN TN FN TN FN TN FN FP TP TP TP TP TP TP TP
0.0
Output of Logistic Regression model
1.0

Exercise (2 min)

Consider a classification model that separates email into two categories: "spam" or "not spam." If you raise the classification threshold, what will happen to precision?

A. Probably increase.

B. Probably decrease.

C. Definitely increase.

D. Definitely decrease.

Consider two models—A and B—that each evaluate the same dataset. Which one of the following statements is true?

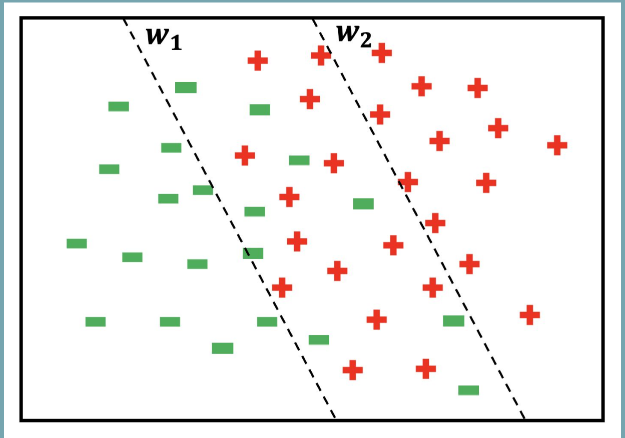
A. If model A has better recall than model B, then model A is better.

B. If model A has better precision and better recall than model B, then model A is probably better.

C. If Model A has better precision than model B, then model A is better.

Recap Exam 1: Modified Logistic Regression

$$\ell_{new-log}(w, x, y) = \begin{cases} \log(1+\exp(-w^T x)) & \text{if } y = 1, \\ 0.01\log(1+\exp(w^T x)) & \text{if } y = -1. \end{cases}$$



What happen to recall?

A. Probably increase.

B. Probably decrease.

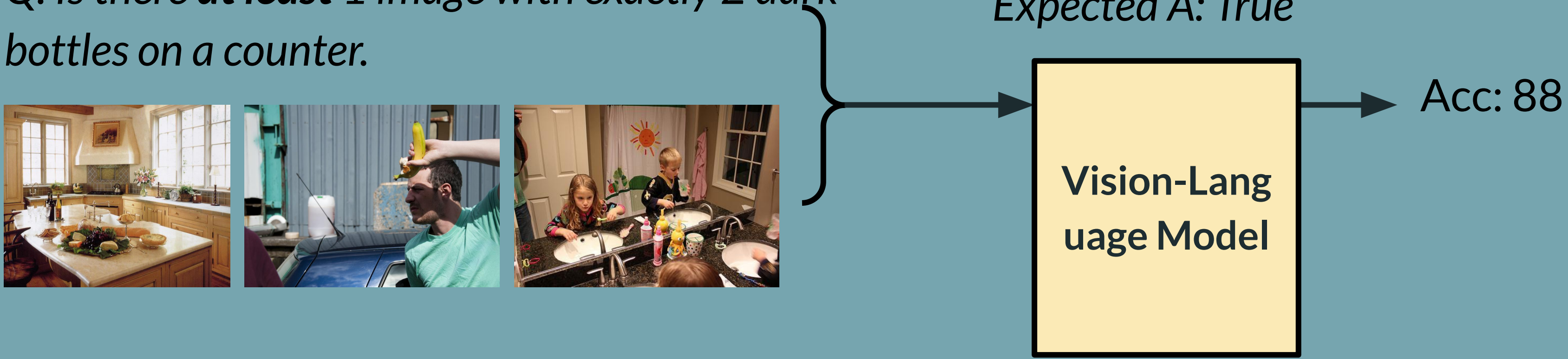
C. Definitely increase.

D. Definitely decrease.

Is w_1 better than w_2 ?

Application: Contrast set

*Q: Is there **at least** 1 image with exactly 2 dark bottles on a counter.*



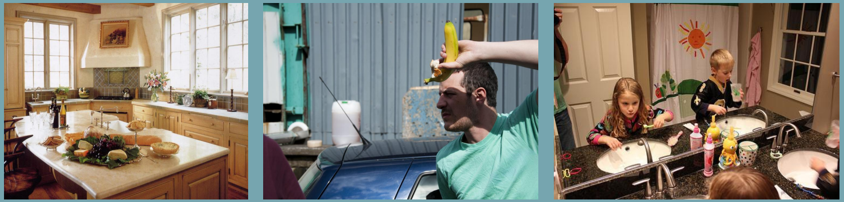
Expected A: True

Vision-Language Model

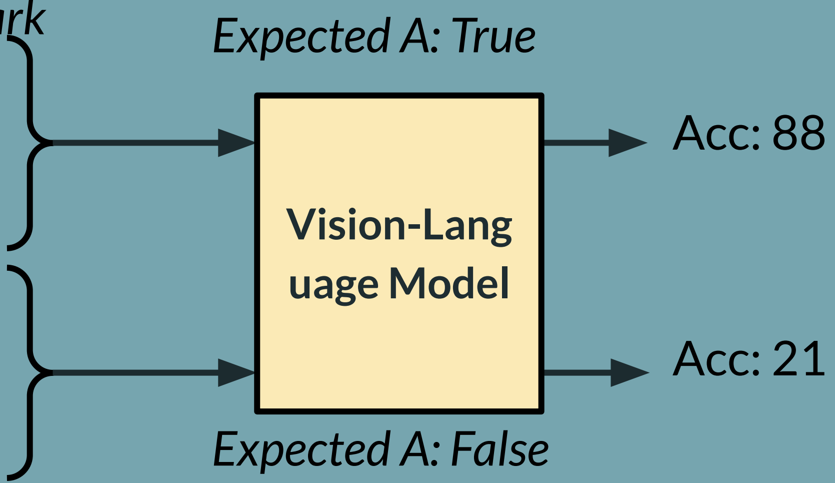
Acc: 88

Application: Contrast set

*Q: Is there **at least** 1 image with exactly 2 dark bottles on a counter.*



*Contrast Q: Is there **less than** 1 image with exactly 2 dark bottles on a counter.*



Expected A: True

Vision-Language Model

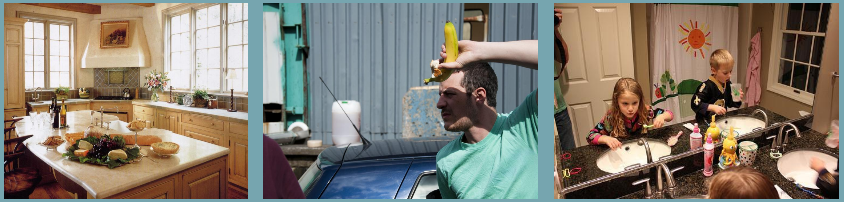
Acc: 88

Acc: 21

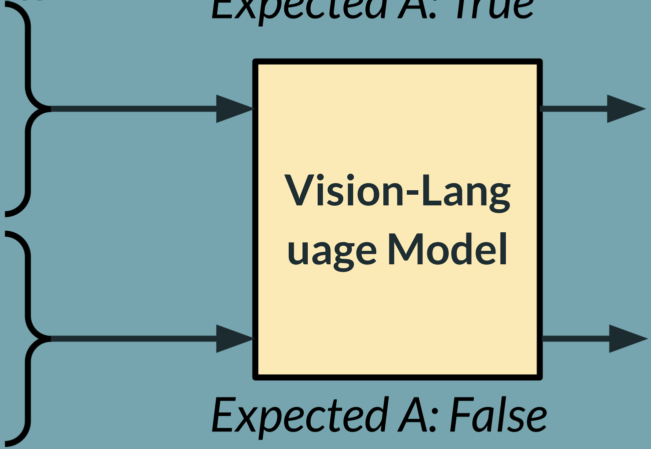
Expected A: False

Application: Contrast set

*Q: Is there **at least** 1 image with exactly 2 dark bottles on a counter.*



*Contrast Q: Is there **less than** 1 image with exactly 2 dark bottles on a counter.*



Expected A: True

Vision-Language Model

Acc: 88

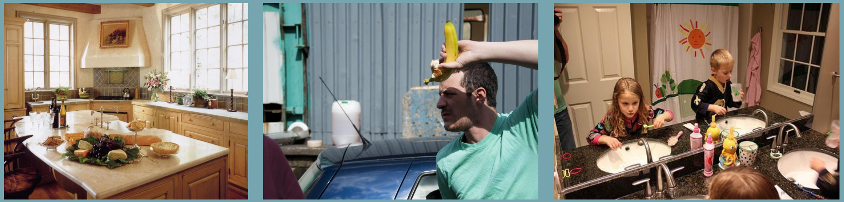
Acc: 21

Expected A: False

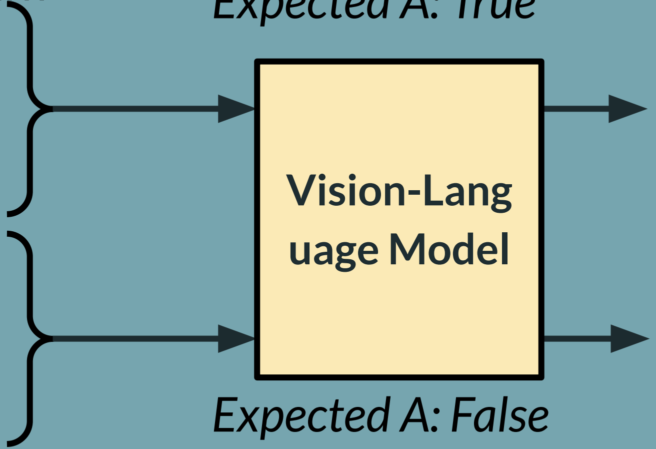
What does this tell us? Contrast Qs are hard? They have low correlation/grounding on images? The VL model is bad?

Application: Contrast set

*Q: Is there **at least** 1 image with exactly 2 dark bottles on a counter.*



*Contrast Q: Is there **less than** 1 image with exactly 2 dark bottles on a counter.*



Expected A: True

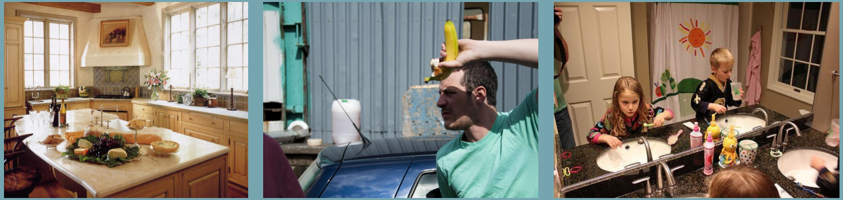
TP: 80	**FP:** 11
FN: 25	**TN:** 188

Expected A: False

TP: 20	**FP:** 83
FN: 157	**TN:** 44

Application: Contrast set

*Q: Is there **at least** 1 image with exactly 2 dark bottles on a counter.*



*Contrast Q: Is there **less than** 1 image with exactly 2 dark bottles on a counter.*

Expected A: True

Vision-Language Model

Expected A: False

TP: 80	**FP:** 11
FN: 25	**TN:** 188

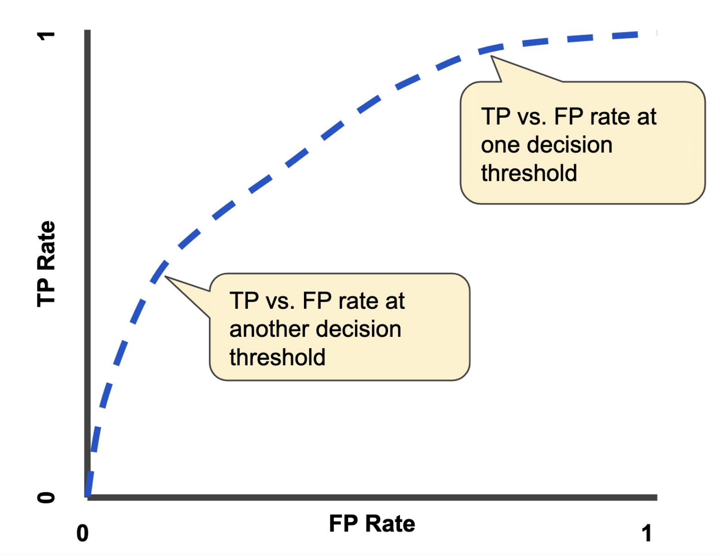
TP: 20	**FP:** 83
FN: 157	**TN:** 44

What does this tell us? (Probably) the model is over-stable on its prediction.

Example from: https://arxiv.org/abs/2210.15037

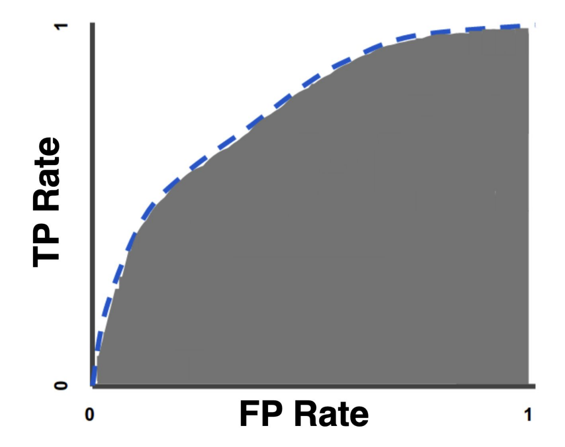
ROC Curve (Receiver Operating Characteristic)

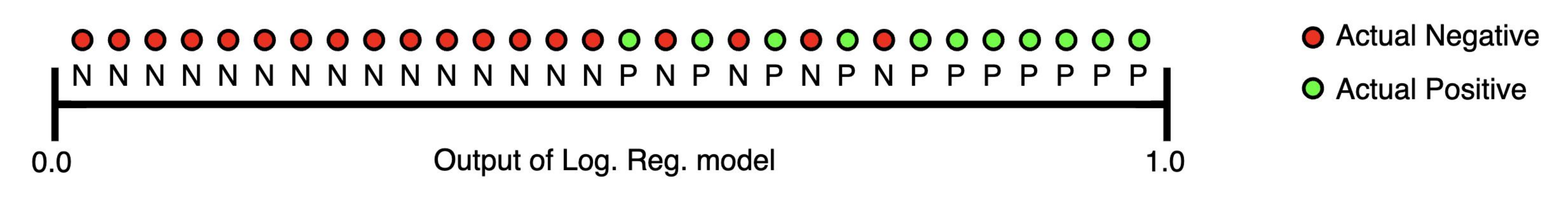
- Each point is the TP and FP rate at one decision threshold
- TPR (Recall) $= \frac{TP}{TP+FN}$
- FPR $= \frac{FP}{FP+TN}$



Evaluation Metrics: AUC (AUROC)

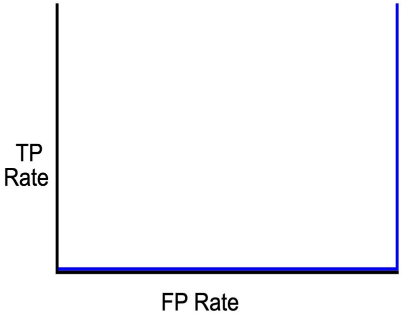
- AUC: “Area under the ROC Curve”
- The probability that the model ranks a random positive example more highly than a random negative example
- Independent of the threshold

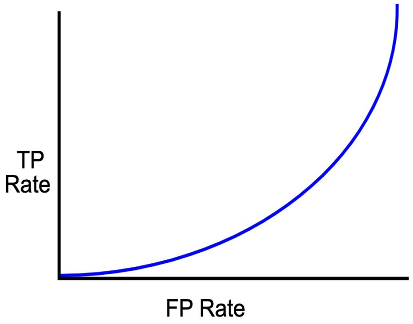


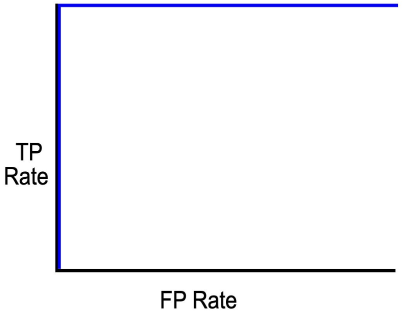

N N N N N N N N N N N N N N N P N P N P N P N P P P P P P P
0.0
Output of Log. Reg. model
1.0
Actual Negative
Actual Positive

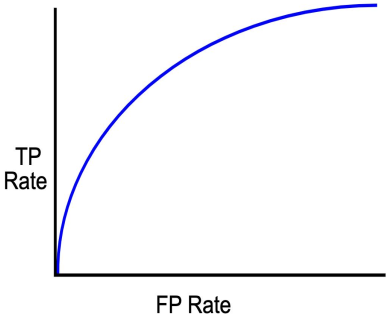
Exercise (2 mins)

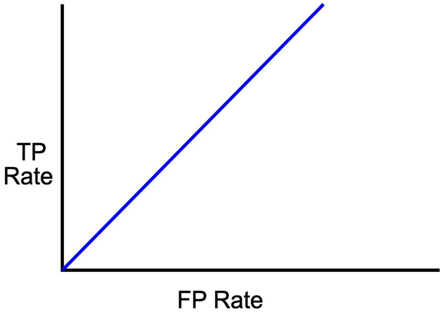
Which of the following ROC curves produce AUC values greater than 0.5?









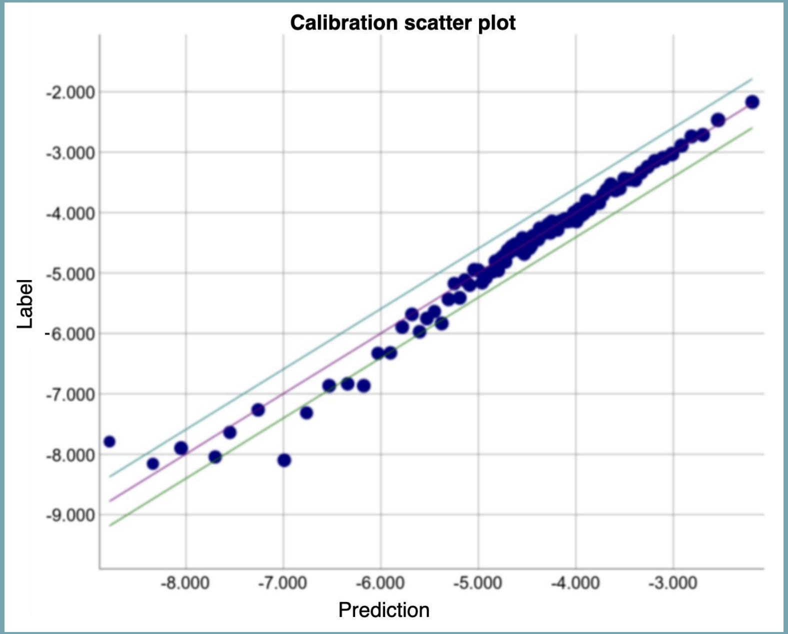


Calibration

- Prediction bias = average of prediction - average of labels
- Calibration layer that adjusts your model's output to reduce the prediction bias
- Possible root causes of prediction bias are:
  - Incomplete feature set
  - Noisy data set
  - Buggy pipeline
  - Biased training sample
  - Overly strong regularization

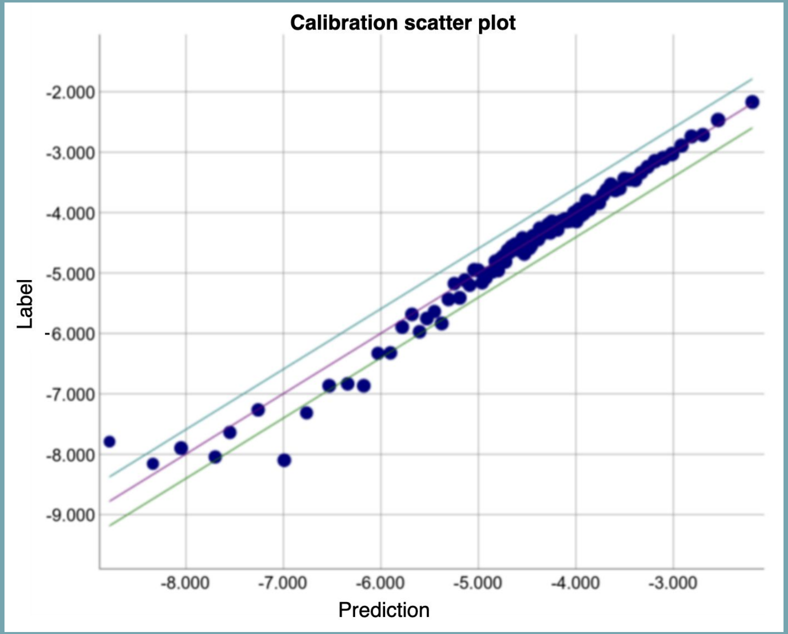
Prediction Bias / Bucketing

- Each dot represents a bucket of 1,000 values. The axes have the following meanings:
- The x-axis - average of values the model predicted for that bucket.
- The y-axis r- average of values in the data set for that bucket.
- Both axes are logarithmic scales.



Prediction Bias / Calibration

- Zero bias alone does not mean everything is perfect
- It’s a great sanity check: incomplete features? noisy data? buggy pipeline?



Evaluation Metric for Others

- Multi-class Confusion Matrix / Evaluation Metric
- OOB Errors
- MSE
- Generative Models
- Unsupervised Learning
- Etc.

Learn from prior work's metric on similar methods / tasks