Google Cloud

A Pinch of Science

Ryan Gillard

Machine Learning on
Google Cloud Platform

The Art of ML

Hyperparameter Tuning

A Pinch of Science

The Science of Neural Networks

Embeddings

Custom Estimator

Google Cloud

Regularization for Sparsity

Ryan Gillard

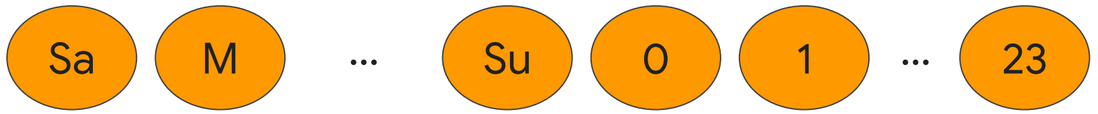
Zeroing out coefficients can help with performance, especially with large models and sparse inputs

Action	Impact
Fewer coefficients to store/load	Reduce memory, model size
Fewer multiplications needed	Increase prediction speed

$$L(w, D) + \lambda \sum^{n} |w|$$

L2 regularization only makes weights small, not zero

Feature crosses lead to lots of input nodes, so having zero weights is especially important

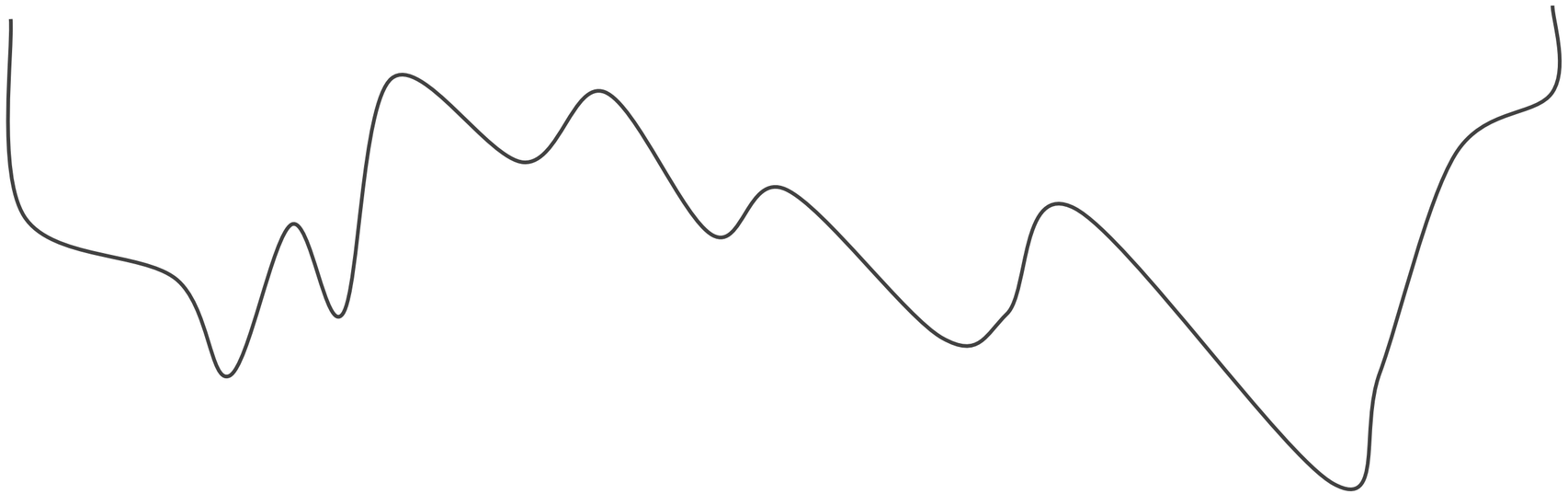


7+24=31 inputs

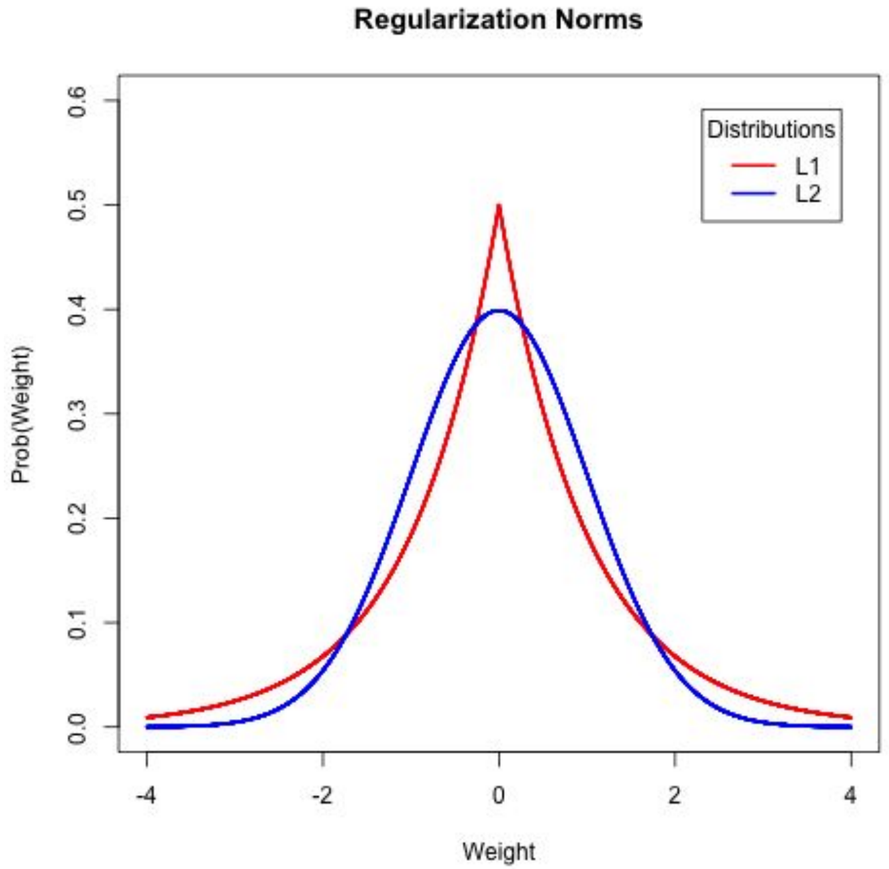
After feature engineering becomes ...

7*24=168 inputs!

L0-norm (the count of non-zero weights) is an NP-hard, non-convex optimization problem



L_1 norm (sum of absolute values of the weights) is convex and efficient; it tends to encourage sparsity in the model



There are many possible choices of norms

$$||x||_p = \left(\sum_{i=1}^{n} |x_i|^p \right)^{1/p}$$

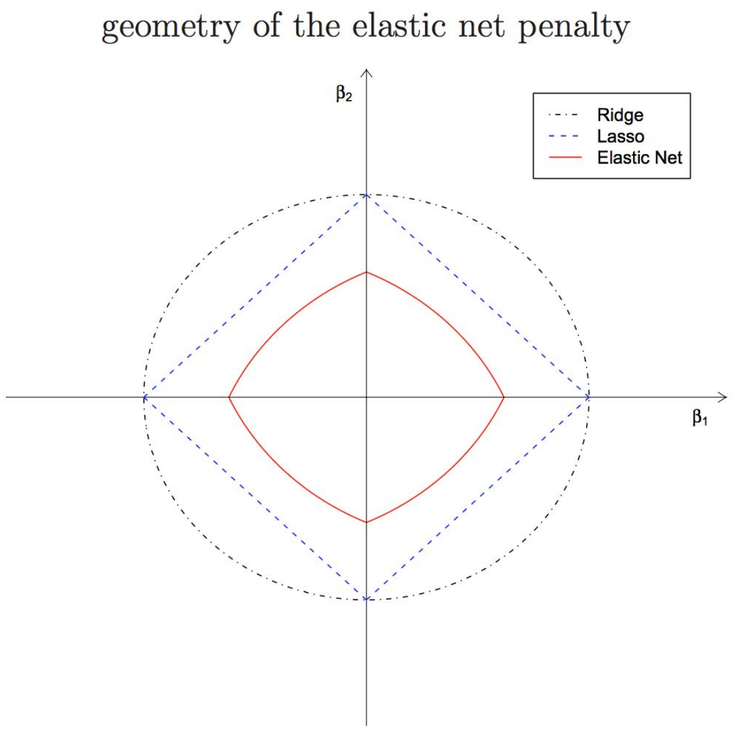
$$L_0\ norm = ||x||_0 = \sum_{i=1}^{n} |x_i|^0$$

$$L_1\ norm = ||x||_1 = \sum_{i=1}^{n} |x_i|$$

$$L_2\ norm = ||x||_2 = \left(\sum_{i=1}^{n} |x_i|^2 \right)^{1/2}$$

$$L_\infty\ norm = ||x||_\infty = max\left\{|x_1|, ..., |x_n|\right\}$$

Elastic nets combine the feature selection of L1 regularization with the generalizability of L2 regularization



$$L(w, D) + \lambda_1 \sum^{n} |w| + \lambda_2 \sum^{n} w^2$$

L1 Regularization Quiz

What does L1 regularization tend to do to a model's low predictive features' parameter weights?

A. Have small magnitudes
B. Have all positive values
C. Have zero values
D. Have large magnitudes

Lab

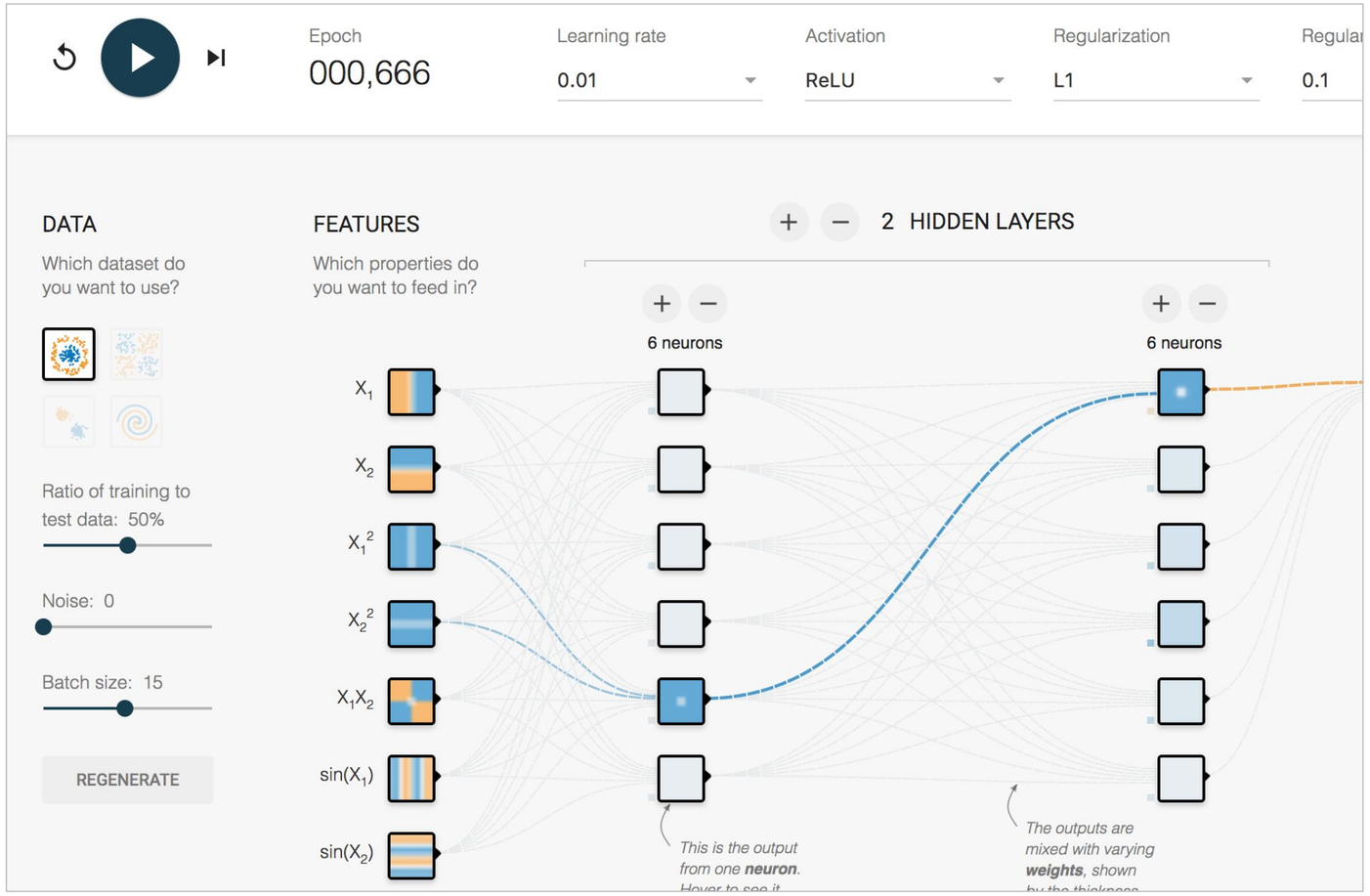
L1 Regularization

Ryan Gillard

Lab: L1 Regularization

https://goo.gl/281mPF

Try with and without L1 regularization.
What's the difference?

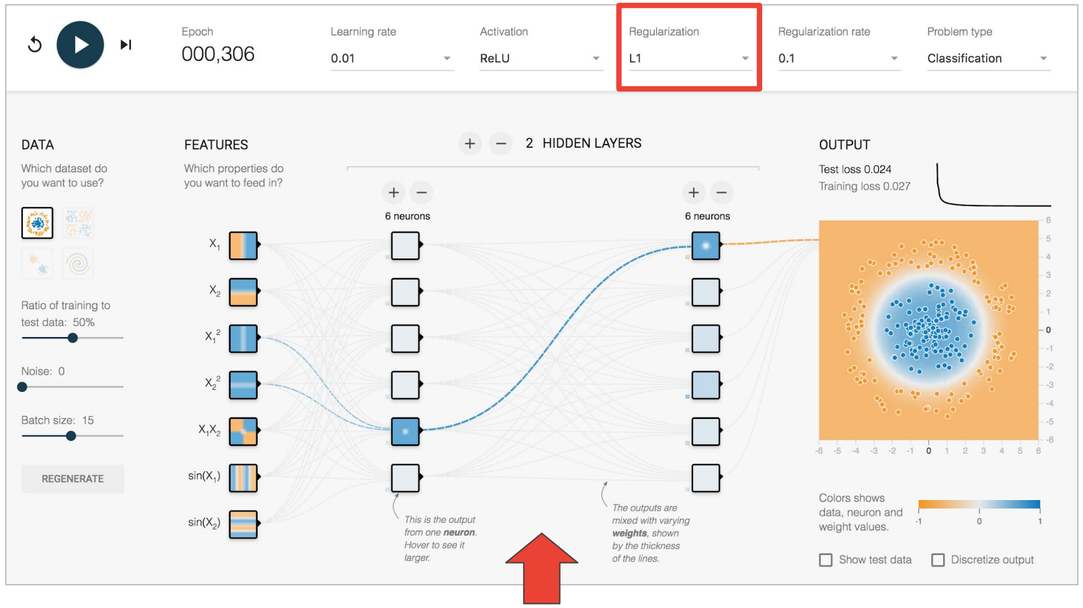


Lab: L1 Regularization

Camtasia

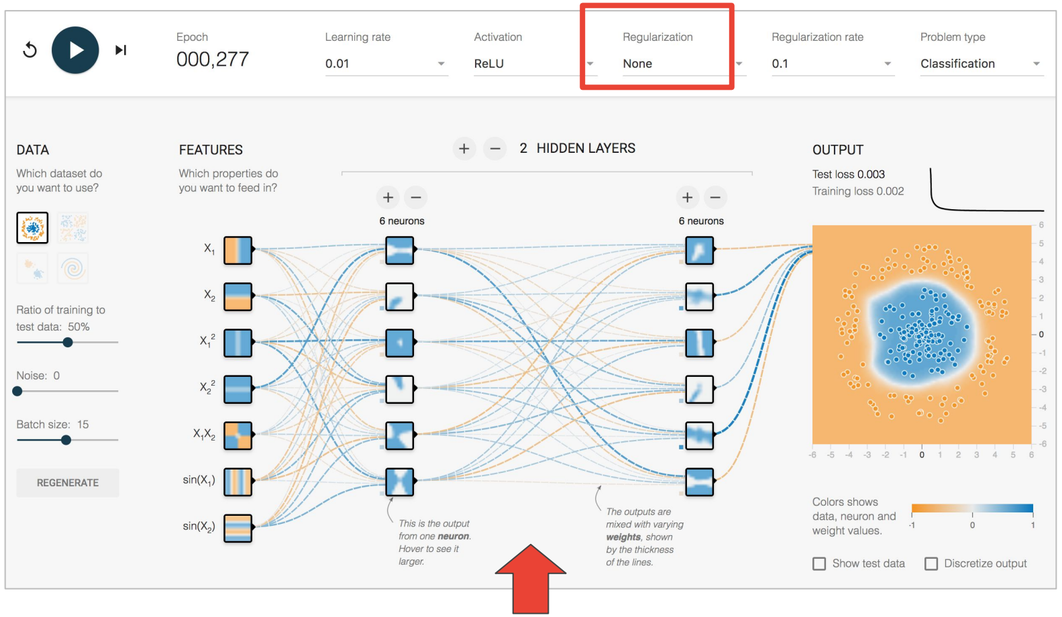
Lab Review: L1 Regularization

With L1 Regularization



Note that with L1 regularization, the useless features all go to zero.

Without L1 Regularization



Without L1 regularization, they retain some value.

Google Cloud

Logistic Regression

Ryan Gillard

Suppose you use linear regression to predict coin flips

You might use features like angle of bend, coin mass, etc. What could go wrong?

Suppose you use linear regression to predict coin flips

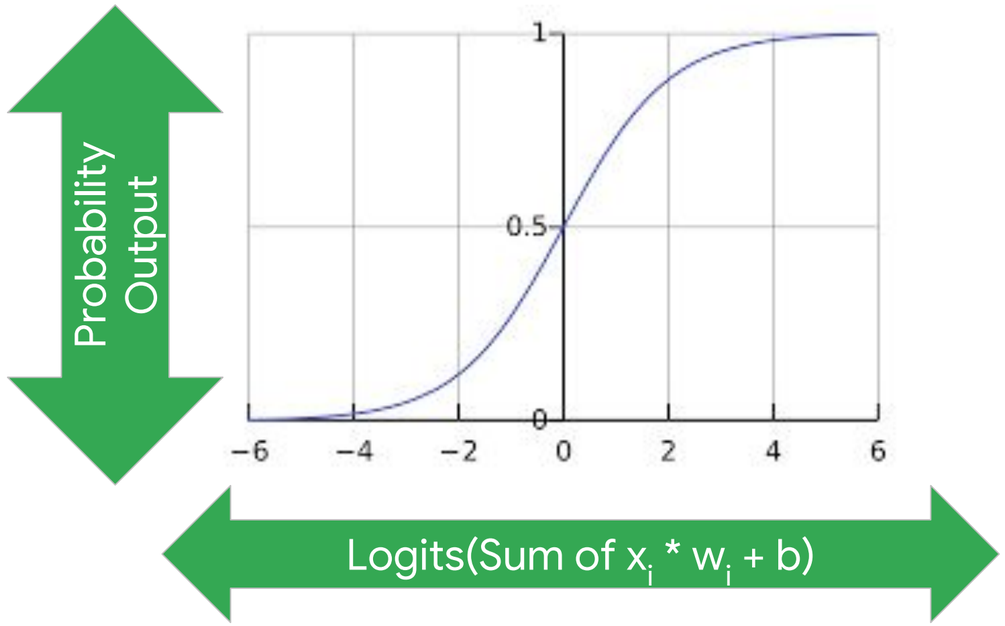
What could go wrong?

Logistic Regression: transform linear regression by a sigmoid activation function

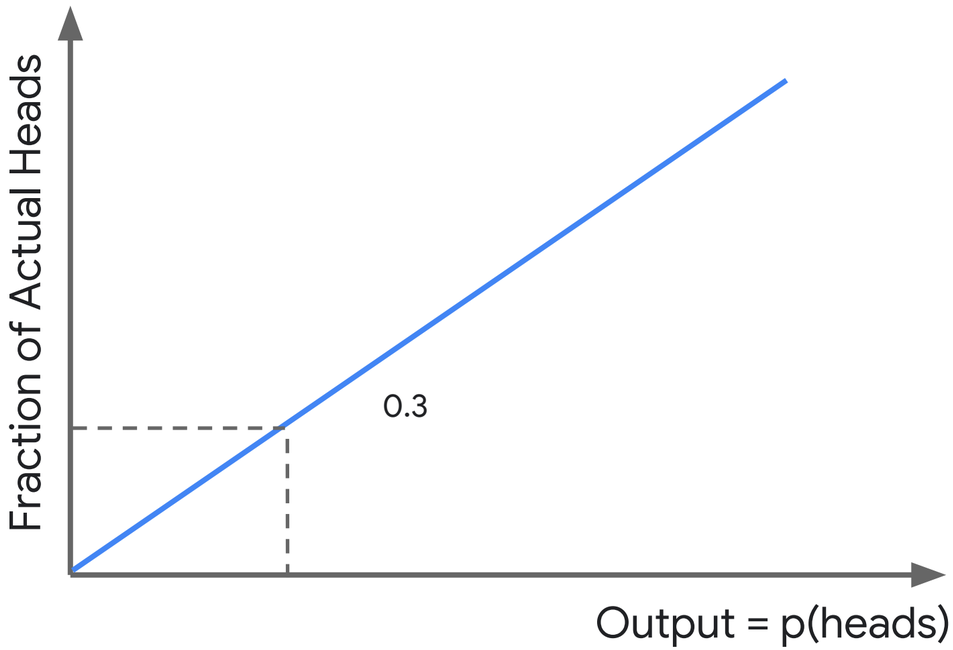
linear model

$$\hat{y} = \frac{1}{1 + e^{-(\mathbf{w}^T\mathbf{x}+b)}}$$

squish through a sigmoid



The output of Logistic Regression is a calibrated probability estimate



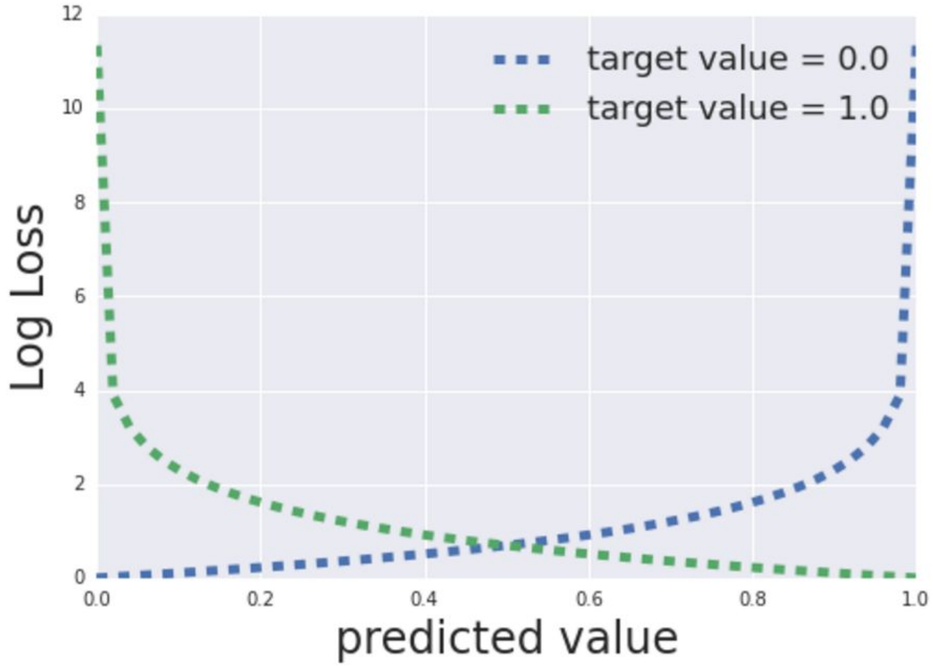
Useful because we can cast binary classification problems into probabilistic problems:

Will customer buy item?

becomes

Predict the probability that customer buys item

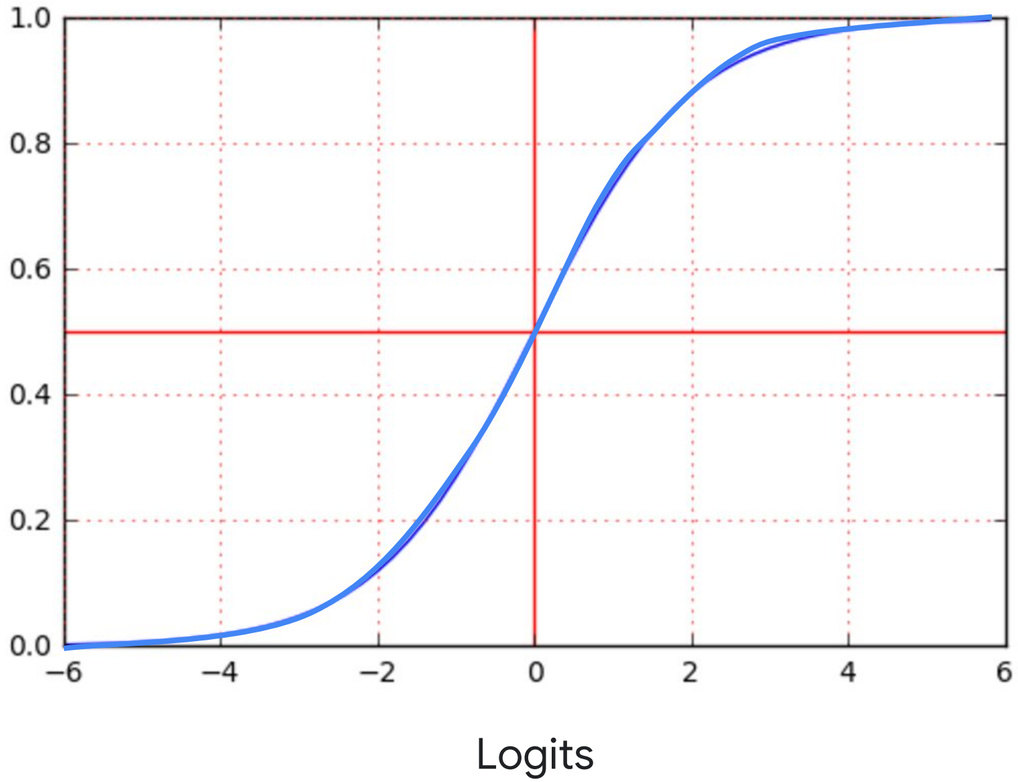
Typically, use cross-entropy (related to Shannon's information theory) as the error metric



Less emphasis on errors where the output is relatively close to the label.

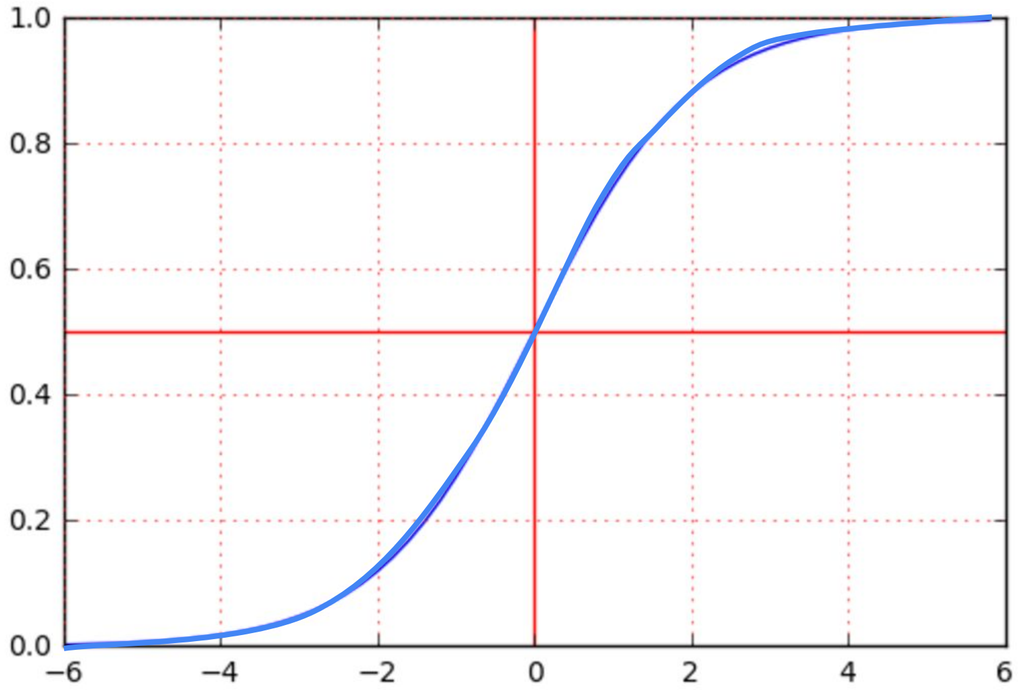
$$LogLoss = \sum_{(\boldsymbol{x},y)\in D} -ylog(\hat{y}) - (1-y)log(1-\hat{y})$$

Regularization is important in logistic regression because driving the loss to zero is difficult and dangerous



Weights will be driven to -inf and +inf the longer we train

Regularization is important in logistic regression because driving the loss to zero is difficult and dangerous



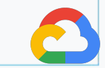
Near the asymptotes, gradient is really small

Weights will be driven to -inf and +inf the longer we train

Logistic Regression Regularization Quiz

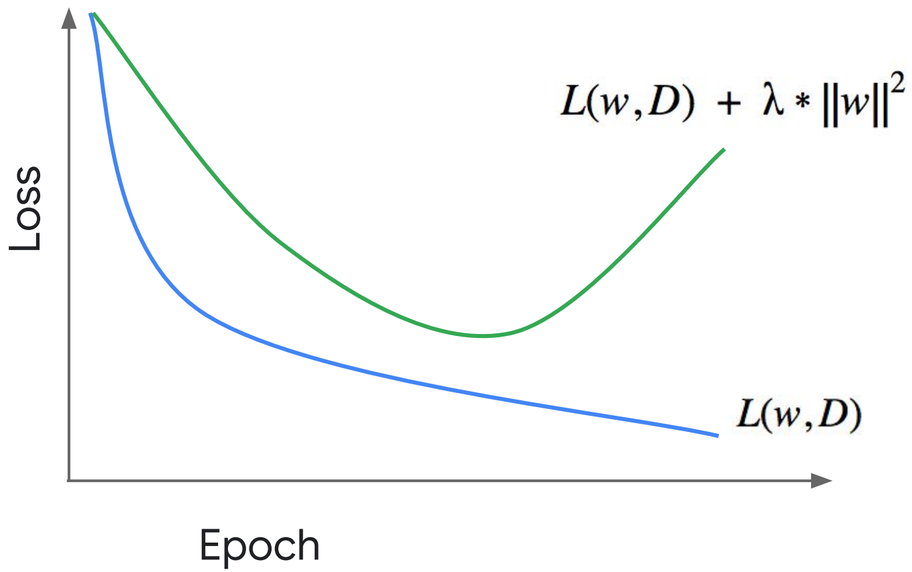
Why is it important to add regularization to logistic regression?

A. Helps stops weights being driven to +/- infinity.
B. Helps logits stay away from asymptotes which can halt training
C. Transforms outputs into a calibrated probability estimate
D. Both A & B
E. Both A & C



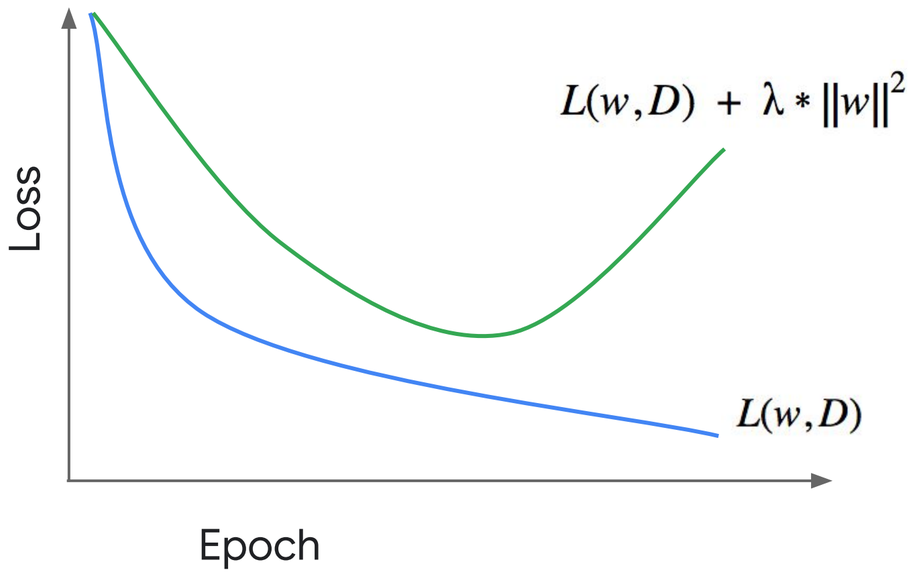
Often we do both regularization and early stopping to counteract overfitting

Regularization

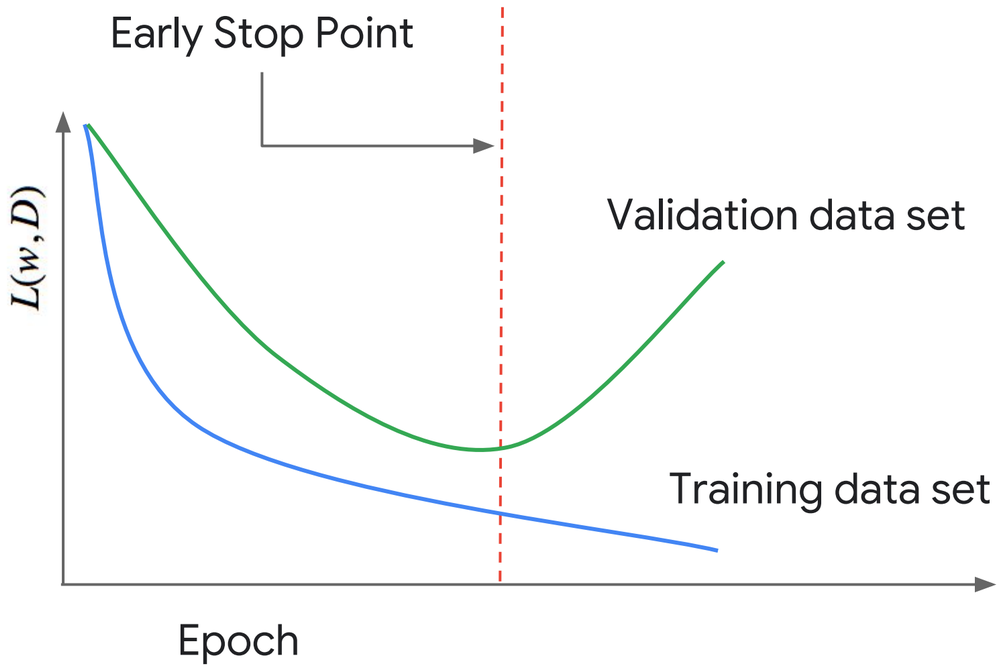


Often we do both regularization and early stopping to counteract overfitting

Regularization



Early stopping



In many real-world problems, the probability is not enough; we need to make a binary decision

Send the mail to spam folder or not?

Approve the loan or not?

Which road should we route the user through?

In many real-world problems, the probability is not enough; we need to make a binary decision

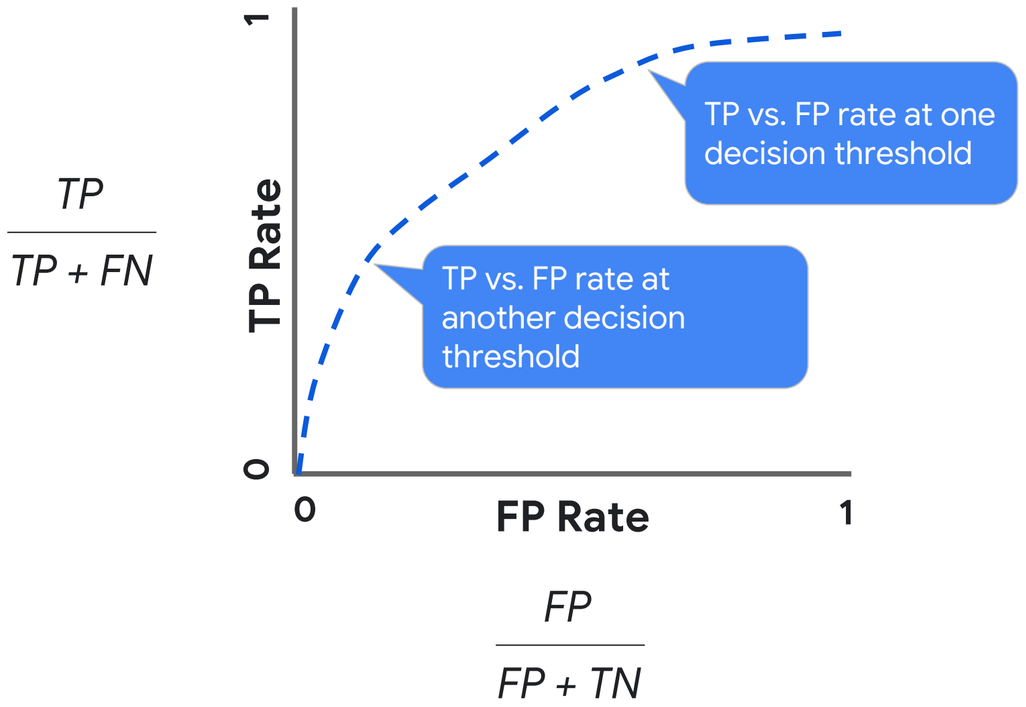
Send the mail to spam folder or not?

Approve the loan or not?

Which road should we route the user through?

Choice of threshold is important and can be tuned

Use the ROC curve to choose the decision threshold based on decision criteria



The Area-Under-Curve (AUC) provides an aggregate measure of performance across all possible classification thresholds

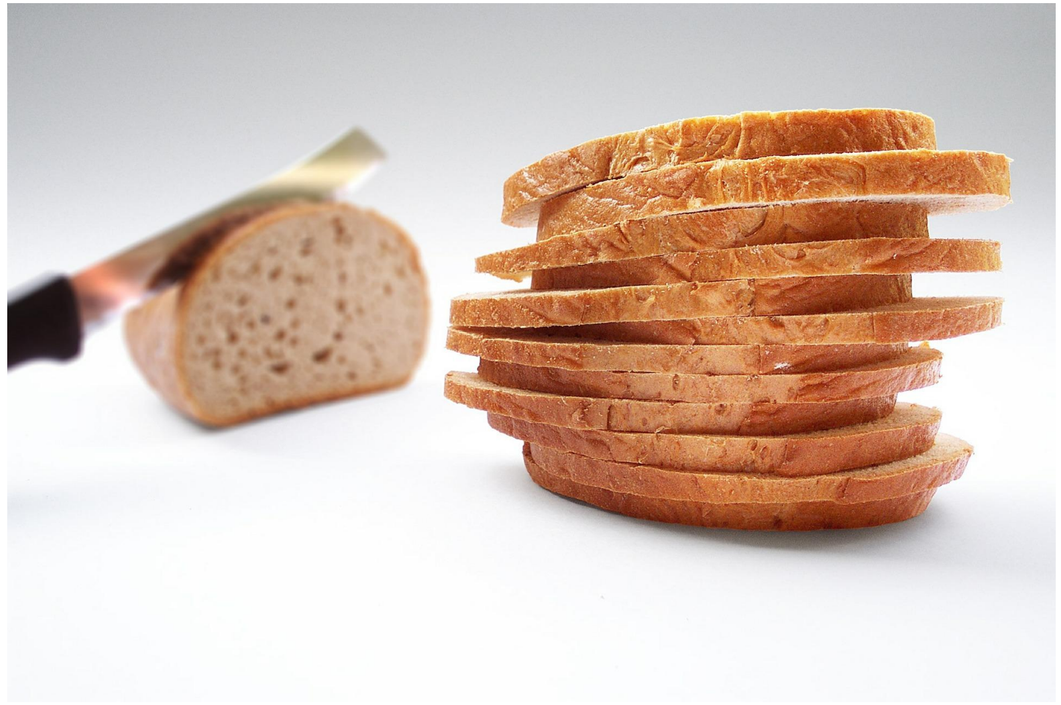
AUC helps you choose between models when you don't know what decision threshold is going to be ultimately used.

"If we pick a random positive and a random negative, what's the probability my model scores them in the correct relative order?"

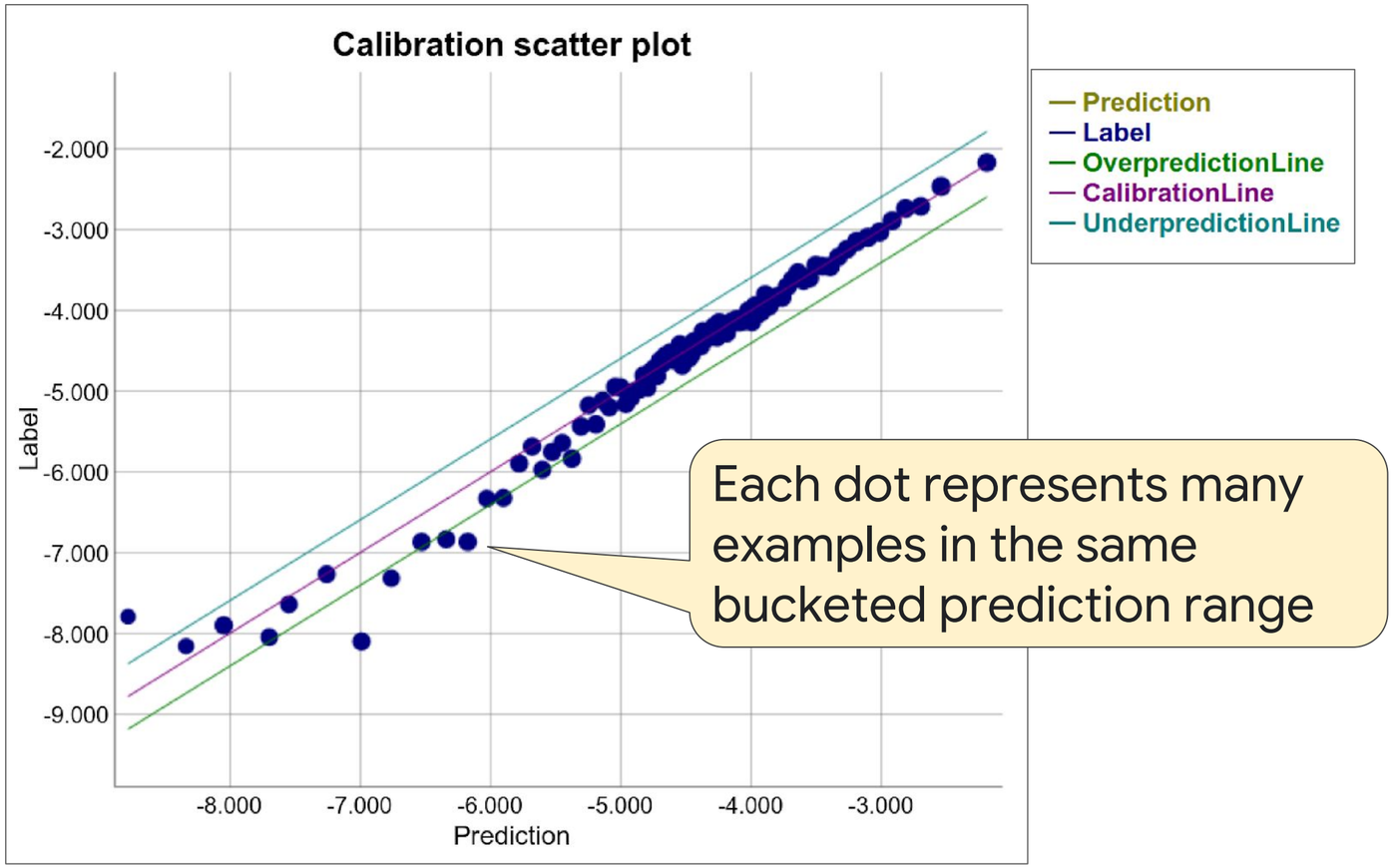
Logistic Regression predictions should be unbiased

average of predictions == average of observations

Look for bias in slices of data, this can guide improvements.



Use calibration plots of bucketed bias to find slices where your model performs poorly



Logistic Regression Quiz

Which of these is important when performing logistic regression?

A. Adding regularization
B. Choosing a tuned threshold
C. Checking for bias
D. All of the above

cloud.google.com