Machine Learning Foundations

Lab 4

Today’s Agenda

Icebreaker (15 mins)

Week 4 Overview + Q&A (30 mins)

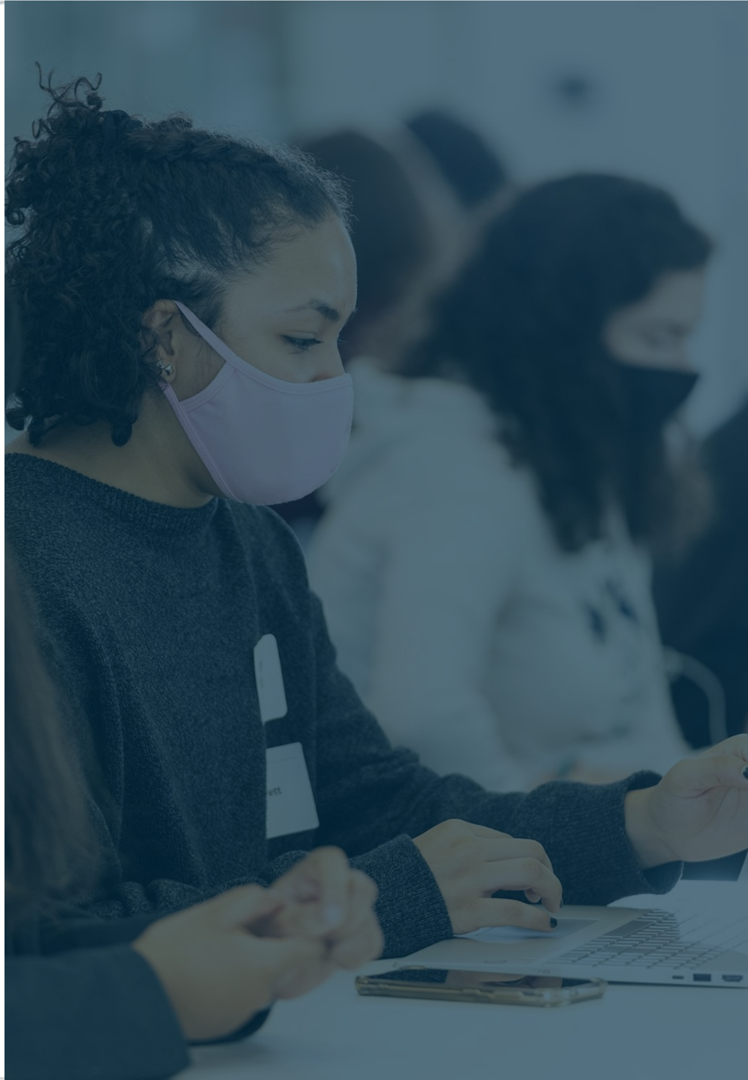
Breakout Group: Big Picture Questions (20 mins)

Class discussion (10 mins)

Break (10 mins)

Breakout Groups: Lab Assignment Working (80 mins)

Concluding Remarks and Survey (15 mins)



Icebreaker “Guess the Outcome”

Instructions

Objective: Your goal is to predict the outcome based on the given features.

- You will be shown a slide with a set of features
- **Analyze the features** and make a prediction about the outcome.

Scenario 1: Predicting Credit Card Fraud

A financial institution is using logistic regression to identify potential credit card fraud. Mark, a 28-year-old male credit card holder, had an unusual transaction of $500 at 3:00 AM from a different country, deviating from his regular spending patterns.

- Age: 28 🎂
- Gender: Male 👨
- Transaction Amount: $500 💳
- Time of Transaction: 3:00 AM ⏰
- Country of Origin: Different from Usual 🌍
- Distance between Transaction Location and Cardholder's Home Address: 500 miles 🗺️
- Time since Last Large Transaction: 6 months 📆

Outcome 1: Fraud

Outcome 2: Not Fraud

Scenario 1: Predicting Credit Card Fraud

A financial institution is using logistic regression to identify potential credit card fraud. Mark, a 28-year-old male credit card holder, had an unusual transaction of $500 at 3:00 AM from a different country, deviating from his regular spending patterns.

- Age: 28 🎂
- Gender: Male 👨
- Transaction Amount: $500 💳
- Time of Transaction: 3:00 AM ⏰
- Country of Origin: Different from Usual 🌍
- Distance between Transaction Location and Cardholder's Home Address: 500 miles 🗺️
- Time since Last Large Transaction: 6 months 📆

Outcome 1: Fraud

Outcome 2: Not Fraud

Scenario 2: Predicting Customer Churn

Café owners are using logistic regression to predict whether Emily, with her average monthly spend of $100 at the cafe, 2-year tenure, and 3 complaints, will continue her loyal patronage or stop being a customer.

- Age: 45 🎂
- Gender: Female 👩
- Average Monthly Spend: $100 💰
- Customer Tenure: 2 years 🕒
- Number of Complaints: 3 ⚠️
- Average Time Spent on Customer Support Calls: 30 minutes ⌛
- Frequency of Product Usage in the Last Month: 5 times 📅

Outcome 1: Stop being a customer

Outcome 2: Retention

Scenario 2: Predicting Customer Churn

Café owners are using logistic regression to predict whether Emily, with her average monthly spend of $100 at the cafe, 2-year tenure, and 3 complaints, will continue her loyal patronage or stop being a customer.

- Age: 45 🎂
- Gender: Female 👩
- Average Monthly Spend: $100 💰
- Customer Tenure: 2 years 🕒
- Number of Complaints: 3 ⚠️
- Average Time Spent on Customer Support Calls: 30 minutes ⏳
- Frequency of Product Usage in the Last Month: 5 times 📅

Outcome 1: Stop being a customer

Outcome 2: Retention

Scenario 3: Predicting Subscription Renewal

Alex has been an active subscriber to a digital platform for 6 months. Alex has logged in 15 times and typically spends 2 hours interacting with the platform during each session. Is Alex likely to renew their subscription?

- Age: 30
- Gender: Non-binary
- Subscription Length: 6 months
- Number of Logins: 15
- Interaction Duration: 2 hours
- Number of Community Forum Contributions: 10
- Participation in Online Events or Webinars: Yes

Outcome 1: Non-Renewal

Outcome 2: Renewal

Scenario 3: Predicting Subscription Renewal

Alex has been an active subscriber to a digital platform for 6 months. Alex has logged in 15 times and typically spends 2 hours interacting with the platform during each session. Is Alex likely to renew their subscription?

- Age: 30
- Gender: Non-binary
- Subscription Length: 6 months
- Number of Logins: 15
- Interaction Duration: 2 hours
- Number of Community Forum Contributions: 10
- Participation in Online Events or Webinars: Yes ☑

Outcome 1: Non-Renewal

Outcome 2: Renewal

Scenario 4: Predicting E-Mail Click-through

At 10AM, Sam received an email with a 10-word subject line with personalized content. What is the likelihood of a click-through?

- Age: 35 🎂
- Gender: Preferred not to say 🤐
- Subject Line Length: 10 words 💬
- Personalization: Yes 🖂
- Time of Sending: 10:00 AM ⏰
- Previous Email Open Rate: 20% 👀
- Interaction with Previous Email Campaigns: Clicked on 3 links 🔗

Outcome 1: Click-through

Outcome 2: Ignore

Scenario 4: Predicting E-Mail Click-through

At 10AM, Sam received an email with a 10-word subject line with personalized content. What is the likelihood of a click-through?

- Age: 35 🎂
- Gender: Preferred not to say 🤐
- Subject Line Length: 10 words 💬
- Personalization: Yes 🖂
- Time of Sending: 10:00 AM ⏰
- Previous Email Open Rate: 20% 👀
- Interaction with Previous Email Campaigns: Clicked on 3 links 🔗

Outcome 1: Click-through

Outcome 2: Ignore

Scenario 5: Predicting Loan Default

Sarah is a 25-year-old unemployed individual who is facing financial hardships. Sarah has a credit score of 500 and a debt-to-income ratio of 40%.

- Age: 25 🎂
- Gender: Female 👩
- Employment Status: Unemployed 🚫💼
- Credit Score: 500 💳
- Debt-to-Income Ratio: 40% 💸
- Educational Background and Qualifications: High school diploma 🎓
- Recent Job Change or Promotion: No ❌

Outcome 1: Repayment

Outcome 2: Default

Scenario 5: Predicting Loan Default

Sarah is a 25-year-old unemployed individual who is facing financial hardships. Sarah has a credit score of 500 and a debt-to-income ratio of 40%.

- Age: 25 🎂
- Gender: Female 👩
- Employment Status: Unemployed 🚫 💼
- Credit Score: 500 💳
- Debt-to-Income Ratio: 40% 💸
- Educational Background and Qualifications: High school diploma 🎓
- Recent Job Change or Promotion: No ❌

Outcome 1: Repayment

Outcome 2: Default

Conclusion

- Logistic regression is a powerful tool solving binary classification problems using computers 🎯
- Specific features and their relationships contribute to predictions 🧩
- Logistic regression is just one of many machine learning algorithms available. It is essential to consider the context, feature selection, and model evaluation when applying these techniques to real-world problems. 🤓

Week 4 Overview + Q&A

Week 4 Overview

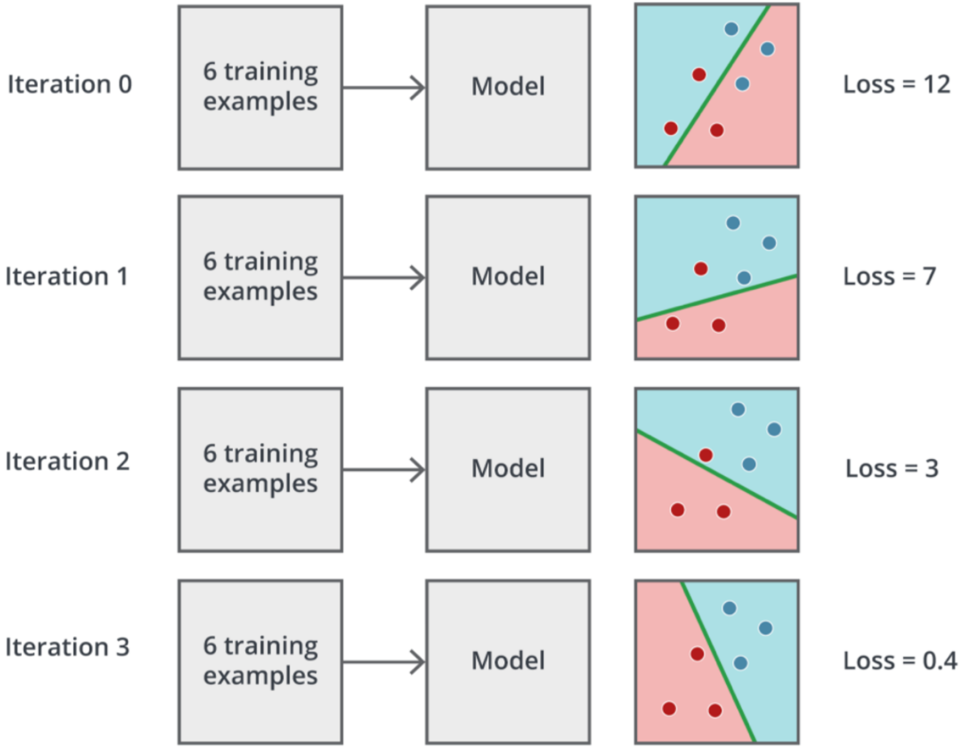
This week covered a number of topics. To refresh your memory, here is what you've completed:

- Analyze the mechanics of logistic regression
- Understand the purpose of using gradient descent and loss functions
- Explore common hyperparameters for logistic regression
- Define the core math concepts required to solve common machine learning problems
- Use NumPy to perform vector and matrix operations
- Explore how linear regression works to solve real world regression problems

Major concepts of a machine learning model

- Model specification – define the model $y = f(x,W)$, how to compute y given x
- Loss function – Given the model-derived $\hat{y}$ and the ground truth y, how to evaluate the difference $L(\hat{y}, y)$
- Model training – Learn the parameters W in the model specification by minimizing the training loss
- Prediction – compute $\hat{y}$ for the x_test
- Model evaluation – Compare the predicted $\hat{y}$ with the ground truth

Using Loss Functions When Training a Linear Model



Iteration 0 — 6 training examples → Model — Loss = 12

Iteration 1 — 6 training examples → Model — Loss = 7

Iteration 2 — 6 training examples → Model — Loss = 3

Iteration 3 — 6 training examples → Model — Loss = 0.4

Loss Functions

In which settings should each of these losses be used?

1. Log loss

$$L_{\mathrm{LL}} = -\frac{1}{N}\sum_{i=1}^{N}\Big(y_i \log(P_i) + (1 - y_i)\log(1 - P_i)\Big)$$

Binary classification problems

1. Mean Squared Error

$$L_{\mathrm{MSE}} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$$

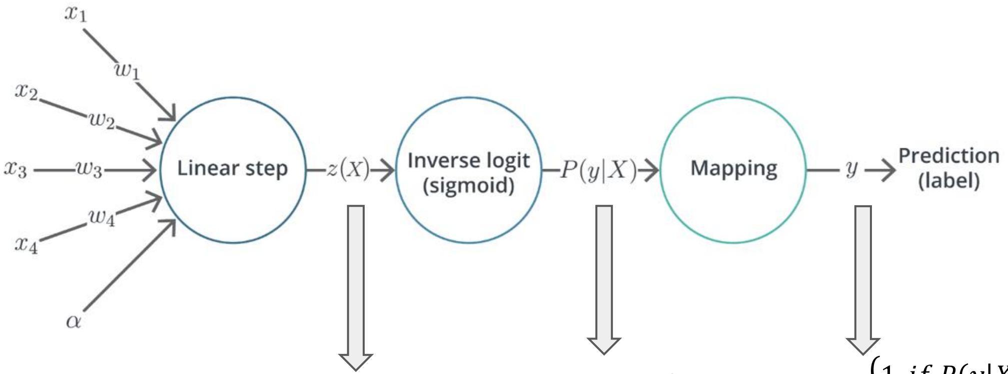
Regression problems

1. Zero-One Loss

$$L_{0/1} = \frac{1}{N}\sum_{i=1}^{N}\mathbb{I}(\hat{y}_i \neq y_i) \qquad \mathbb{I}(\hat{y}_i \neq y_i) = \begin{cases} 1 & \text{if } \hat{y}_i \neq y_i \\ 0 & \text{if } \hat{y}_i = y_i \end{cases}$$

Multiclass classification problems (after model training)

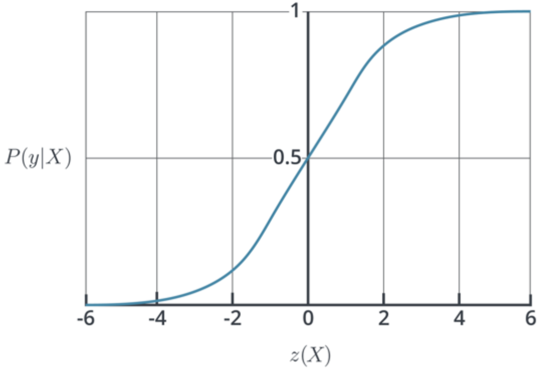
Making a Prediction Using Logistic Regression

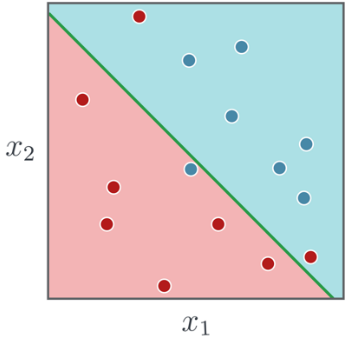


$$z(X) = \alpha + w_1 x_1 + w_2 x_2 + \ldots w_n x_n$$

$$P(y|X) = \frac{1}{1+e^{-z(X)}}$$

$$y = \begin{cases} 1 & if\ P(y|X) > 0.75 \\ 0 & else \end{cases}$$

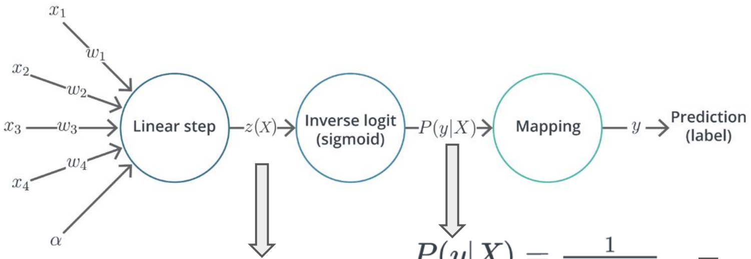




Decision Boundary

Training LR Using Gradient Descent: Finding Weights

Computing t̃ Эự· ¸Ớ

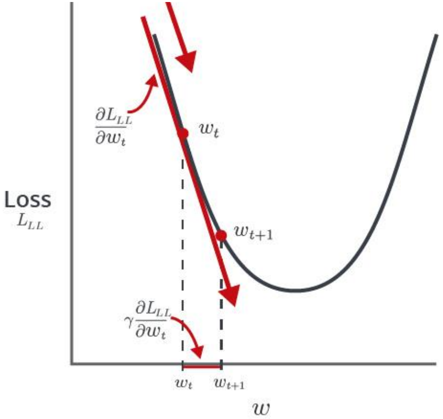


$$z(X) = \alpha + w_1 x_1 + w_2 x_2 + \ldots w_n x_n$$

$$P(y|X) = \frac{1}{1+e^{-z(X)}} \quad = \quad p_i$$

Finding weights:

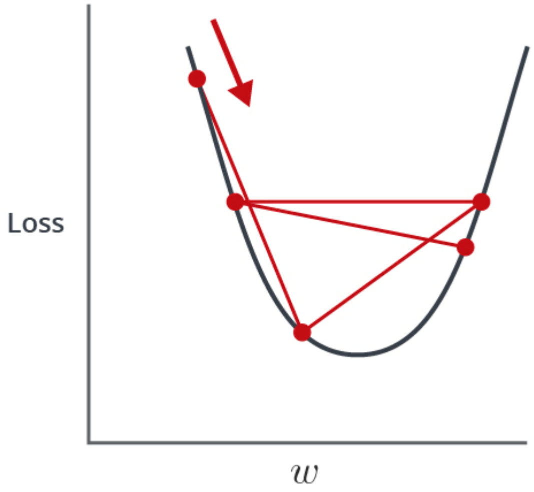
$$L_{\text{LL}} = -\frac{1}{N}\sum_{i=1}^{N}\Big(y_i \log(P_i) + (1-y_i)\log(1-P_i)\Big)$$



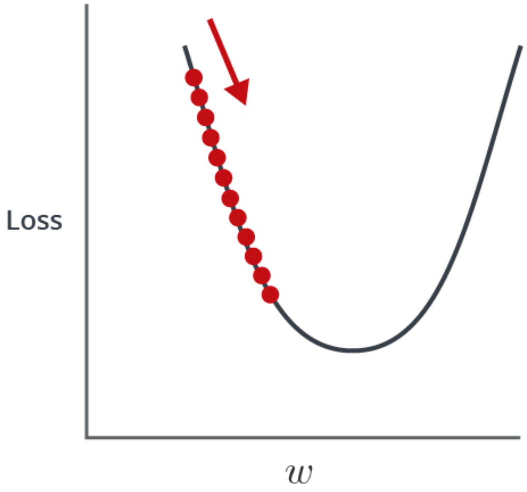
$$w_{t+1} = w_t - \gamma \frac{\partial}{\partial w} L_{LL}(w_t)$$

$$\alpha_{t+1} = \alpha_t - \gamma \frac{\partial}{\partial \alpha} L_{LL}(\alpha_t)$$

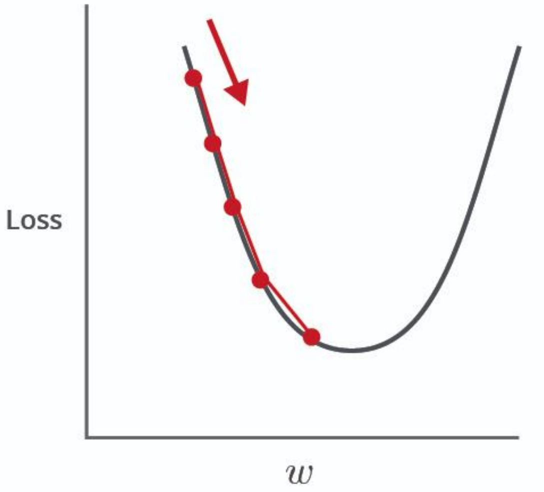
Learning Rate γ



Too high



Too low



Ideal

Determining Learning Rate γ

Gradient Descent:

$$w_{t+1} = w_t - \gamma \cdot \nabla L_{LL}(w_t)$$

$$\gamma = \frac{1}{H(w_t)}$$

Use the Hessian function to compute optimal

- Second derivative of function, represents curvature
- Steep curve → small rate, gradual curve, → larger rate
- For log loss we have:

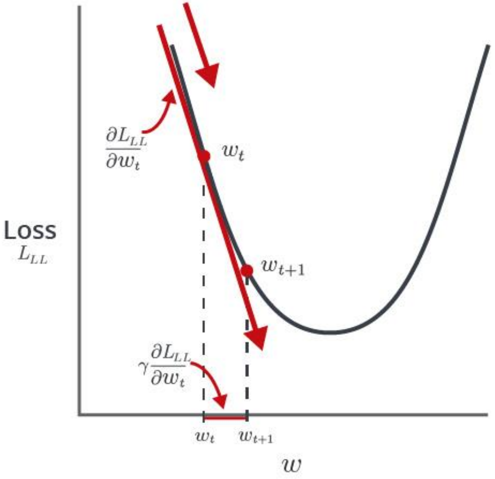
P: output model probabilities from weights at time t

Y: ground truth label vector

X: data matrix

Q: intermediate computation

H: Hessian matrix



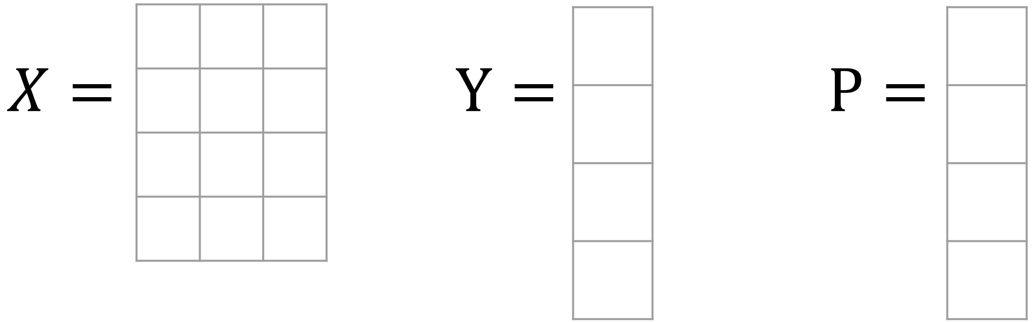
$\frac{\partial L_{LL}}{\partial w_t}$
w_t
Loss
L_{LL}
w_{t+1}
$\gamma\frac{\partial L_{LL}}{\partial w_t}$
w_t
w_{t+1}
w

$$\nabla L_{LL} = -1 * (Y - P) \cdot X$$

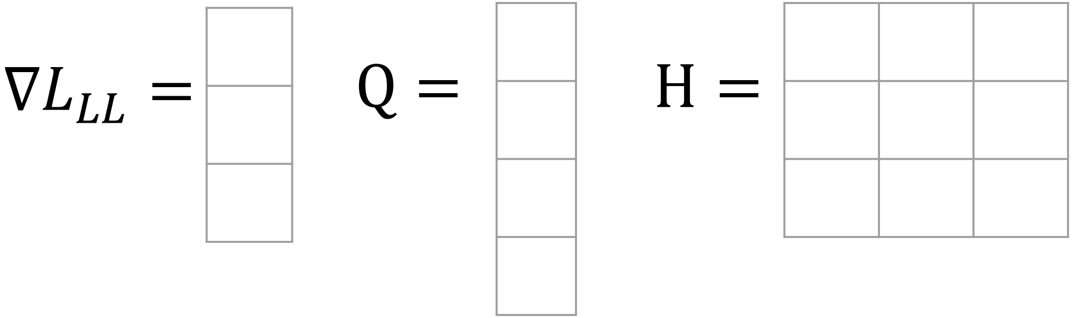
$$Q = P * (1 - P)$$

$$H = (X * Q)^T \cdot X$$

Multiplication, dot product, and matrix multiplication



What are the shapes of ∇L_{LL}, Q, and H?



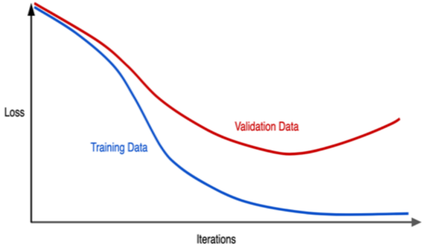
multiplication

Dot product

$$\nabla L_{LL} = -1 * (Y - P) \cdot X$$
$$Q = P * (1 - P)$$
$$H = (X^T * Q) \cdot X$$

Regularization

- During training, our goal should be to minimize loss and avoid overfitting by minimizing the model’ Regularization is a technique that accomplishes this goal by penalizing complex models in an atte overfitting



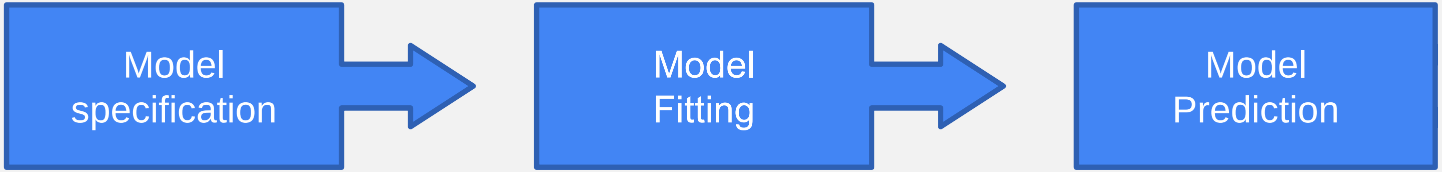
- Two types: L1 and L2
- Hyperparameter C controls how much regularization applied
- Comparison:

L1 Regularization	L2 Regularization
Penalizes the sum of absolute value of weights.	Penalizes the sum of square weights.
It has a sparse solution.	It has non-sparse solution.
It gives multiple solutions.	It has only one solution.
Constructed in feature selection.	No feature selection.
Robust to outliers.	Not robust to outliers.
It generates simple and interpretable models.	It gives more accurate predictions when the output variable is the function of whole input variables.
Unable to learn complex data patterns.	Able to learn complex data patterns.
Computationally inefficient over non-sparse conditions.	Computationally efficient because of having analytical solutions.

Tyagi, Neelam “L2 and L1 Regularization in Machine Learning” *AnalyticsSteps*, 2021, https://www.analyticssteps.com/blogs/l2-and-l1-regularization-machine-learning.

Logistic Regression in code

Scikit-learn’s 3 Steps



Model specification

Model Fitting

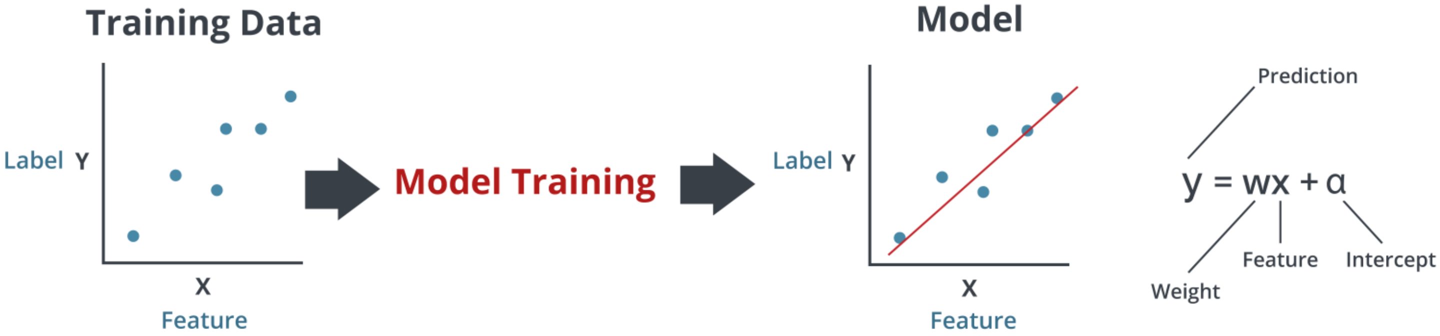
Model Prediction

```
% Initialize the model
model = LogisticRegression(C=regVal)

% Train the model using training sets
model.fit(X_train, y_train)

# Make predictions on the test set
prediction = model.predict(X_test)
```

Linear Regression



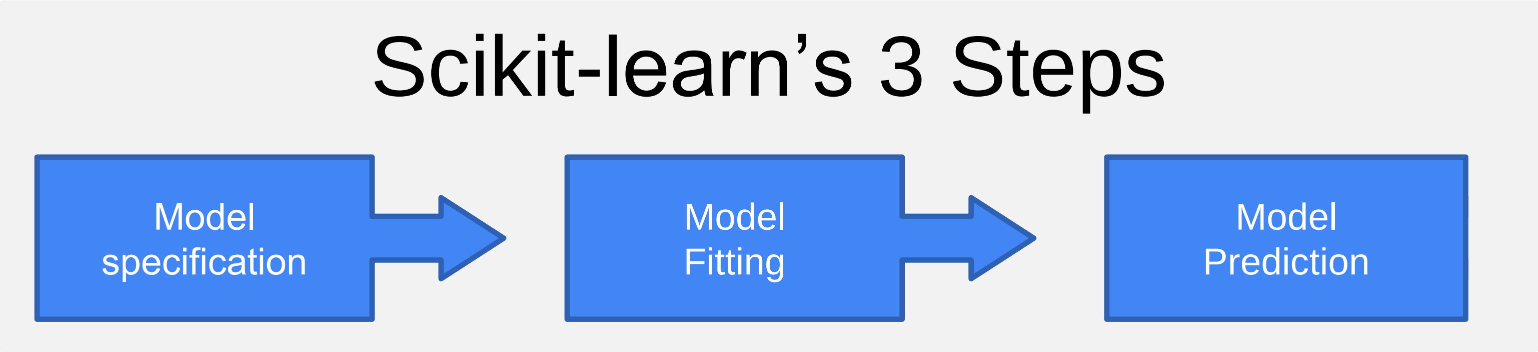
Training Data

Model Training

Model

$y = wx + \alpha$

Linear Regression in code



Scikit-learn’s 3 Steps

Model specification

Model Fitting

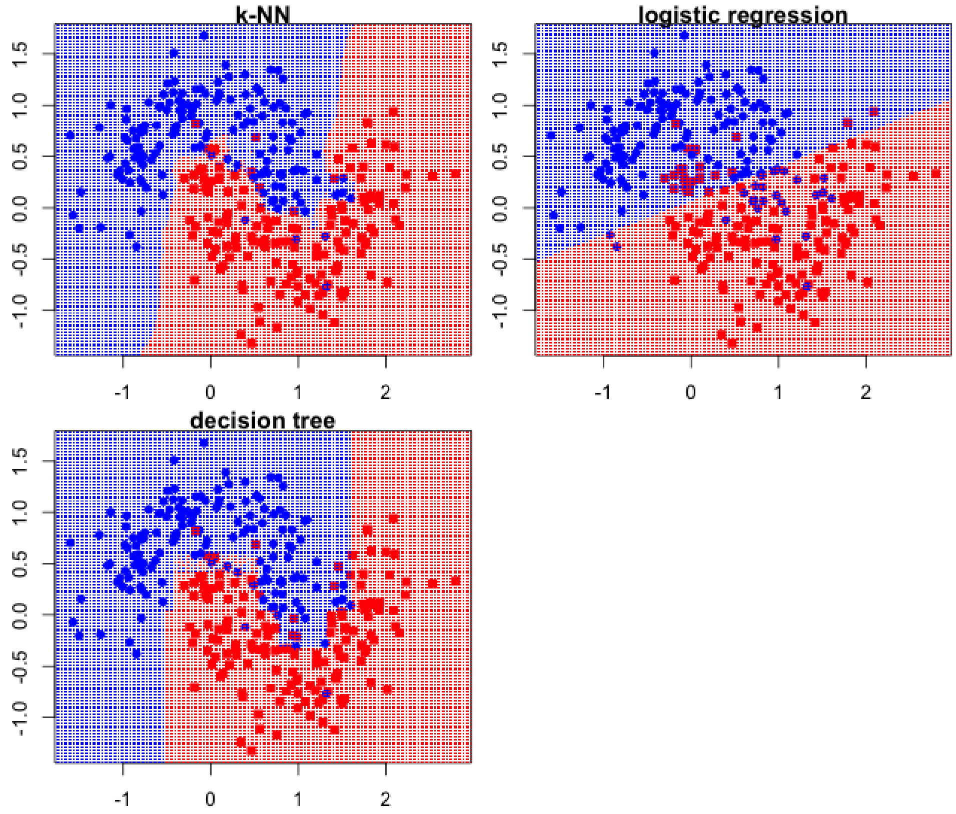
Model Prediction

```
% Initialize the model
model = LinearRegression()

% Train the model using training sets
model.fit(X_train, y_train)

# Make predictions on the test set
prediction = model.predict(X_test)
```

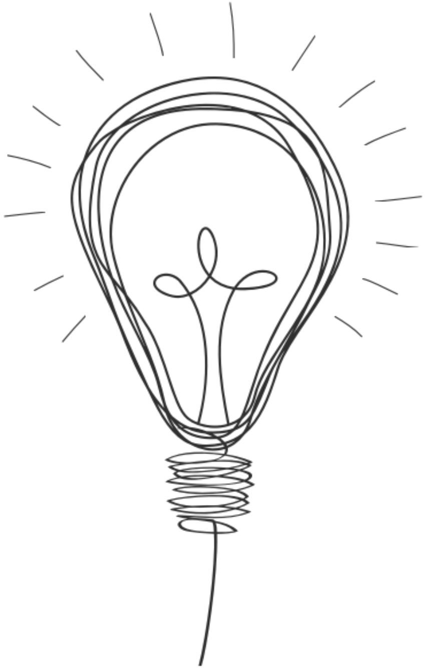
Comparing models for binary classification



Can we apply linear regression to this problem?

Questions & Answers

What questions do you have about the online content this week?



Breakout Groups: Big Picture Questions

Big Picture Questions

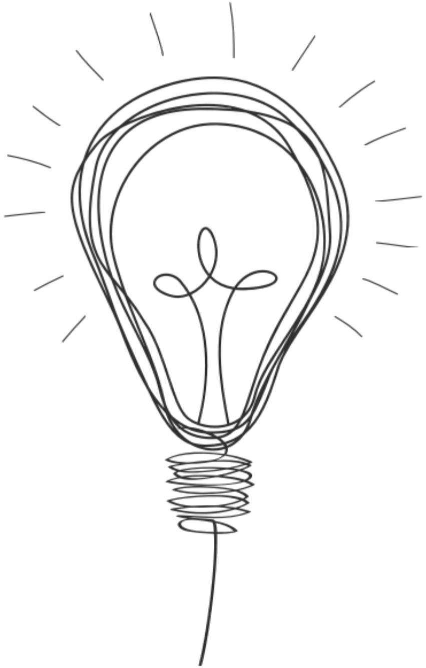
You have 20 minutes to discuss the following questions within your breakout groups:

- How do decision trees and logistic regression differ in terms of their underlying algorithms or models?
- What are the differences in the type of output or predictions generated by decision trees and logistic regression?
- In which scenarios would logistic regression be more suitable or preferable compared to a decision tree?
- What factors or characteristics of the dataset influence selection between logistic regression and decision trees?
- Why is it good or useful to estimate the probability of something occurring, as opposed to just making a binary classification decision?

Class Discussion

Class Discussion: Responses to Big Picture Questions

Let’s hear your classmates’ responses.



Break! (10 minutes)

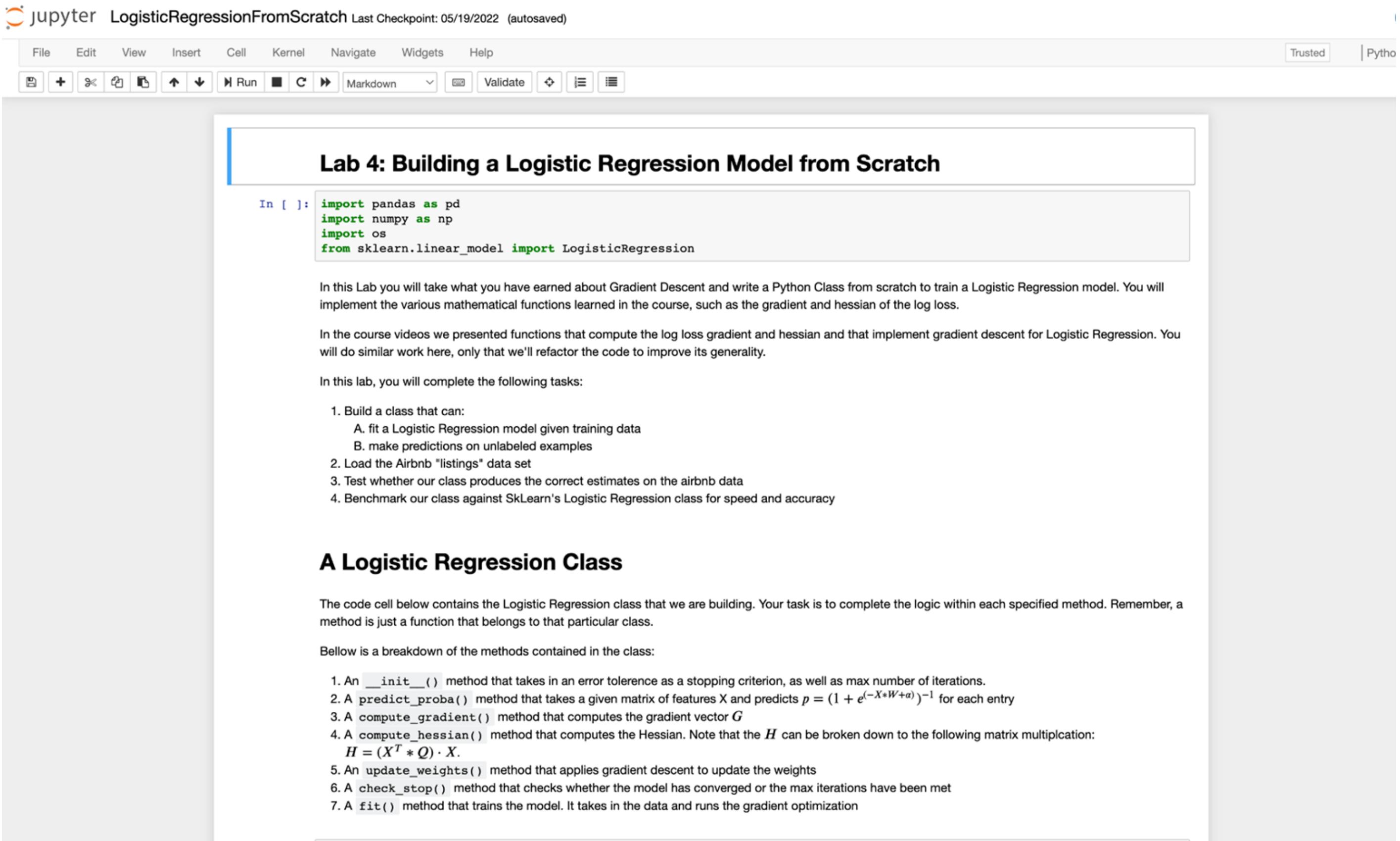
Breakout Groups: Lab Assignment

Lab 4

In this lab, you will:

- Load and split the data into training and test sets
- Write a Python class that will train a logistic regression model
- Compare your implementation to scikit-learn's implementation

Lab 4



Lab 4: Building a Logistic Regression Model from Scratch

In [ ]:

```python
import pandas as pd
import numpy as np
import os
from sklearn.linear_model import LogisticRegression
```

In this Lab you will take what you have earned about Gradient Descent and write a Python Class from scratch to train a Logistic Regression model. You will implement the various mathematical functions learned in the course, such as the gradient and hessian of the log loss.

In the course videos we presented functions that compute the log loss gradient and hessian and that implement gradient descent for Logistic Regression. You will do similar work here, only that we'll refactor the code to improve its generality.

In this lab, you will complete the following tasks:

1. Build a class that can:
   A. fit a Logistic Regression model given training data
   B. make predictions on unlabeled examples
2. Load the Airbnb "listings" data set
3. Test whether our class produces the correct estimates on the airbnb data
4. Benchmark our class against SkLearn's Logistic Regression class for speed and accuracy

A Logistic Regression Class

The code cell below contains the Logistic Regression class that we are building. Your task is to complete the logic within each specified method. Remember, a method is just a function that belongs to that particular class.

Bellow is a breakdown of the methods contained in the class:

1. An `__init__()` method that takes in an error tolerence as a stopping criterion, as well as max number of iterations.
2. A `predict_proba()` method that takes a given matrix of features X and predicts $p = (1 + e^{(-X*W+\alpha)})^{-1}$ for each entry
3. A `compute_gradient()` method that computes the gradient vector G
4. A `compute_hessian()` method that computes the Hessian. Note that the H can be broken down to the following matrix multiplcation: $H = (X^T * Q) \cdot X$.
5. An `update_weights()` method that applies gradient descent to update the weights
6. A `check_stop()` method that checks whether the model has converged or the max iterations have been met
7. A `fit()` method that trains the model. It takes in the data and runs the gradient optimization

Working Session Debrief

Lab Debrief

- What did you enjoy about this lab?
- What did you find hard about this lab?
- What questions do you still have about this lab?
- How did you approach problem-solving during the exercise?
- What would you do differently if you were to repeat the exercise?

Concluding Remarks

Concluding Remarks

Logistic Regression	Decision Trees
• Works well for linearly separable data • Outputs probabilities • Decides class based on what side of a line a data point lies • Simpler and usually • Not prone to overfitting/noise has little impact • Used for classification	• Works well for data with complex relationships between features and output label • Outputs classifications • Decides class based on a number of decisions (greater than or equal to's) • Prone to overfitting/majorly effected by noise • Used for classification or regression • Works well with categorical features • More interpretable

Next week

In the following week, you will:

- Understand the importance of model selection in machine learning
- Choose model evaluation metrics that are appropriate for the application
- Choose appropriate model candidates and hyperparameters for testing
- Set up training/validation/test splits for model selection
- Apply feature selection techniques to get a better-performing model

And in the lab, you will:

- Build a logistic regression model using scikit-learn's default hyperparameter value for C
- Find the optimal logistic regression model using GridSearchCV
- Evaluate both models' predictions using a confusion matrix
- Plot the precision-recall curve, ROC, and compute the AUC for both models
- Practice the SelectKBest feature selection method

Content + Lab Feedback Survey

Content + Lab Feedback Survey

To complete your lab, please answer the following questions about BOTH your online modules and your lab experience. Your input will help pay it forward to the Break Through Tech student community by enabling us to continuously improve the learning experience that we provide to our community.

Thank you for your thoughtful feedback!

https://forms.gle/xdfN3Vy1BYMUHFvu8