Midterm 1 Prep

CPSC 330 Tutorial 4

Introduction to Machine Learning

Which of the following problems is **most suitable** for machine learning?

A) Computing the sum of two numbers
B) Predicting housing prices based on historical data
C) Sorting a list of numbers
D) Checking if a number is prime

Introduction to Machine Learning

Which of the following problems is **most suitable** for machine learning?

A) Computing the sum of two numbers
B) Predicting housing prices based on historical data
C) Sorting a list of numbers
D) Checking if a number is prime

Introduction to Machine Learning

Unsupervised learning would be more useful for:

A) Predicting stock prices
B) Classifying spam emails
C) Identifying types of customers of an online retail store
D) Diagnosing diseases

Introduction to Machine Learning

Unsupervised learning would be more useful for:

A) Predicting stock prices
B) Classifying spam emails
C) Identifying types of customers of an online retail store
D) Diagnosing diseases

Introduction to Machine Learning

Which of the following is an example of a **regression** problem?

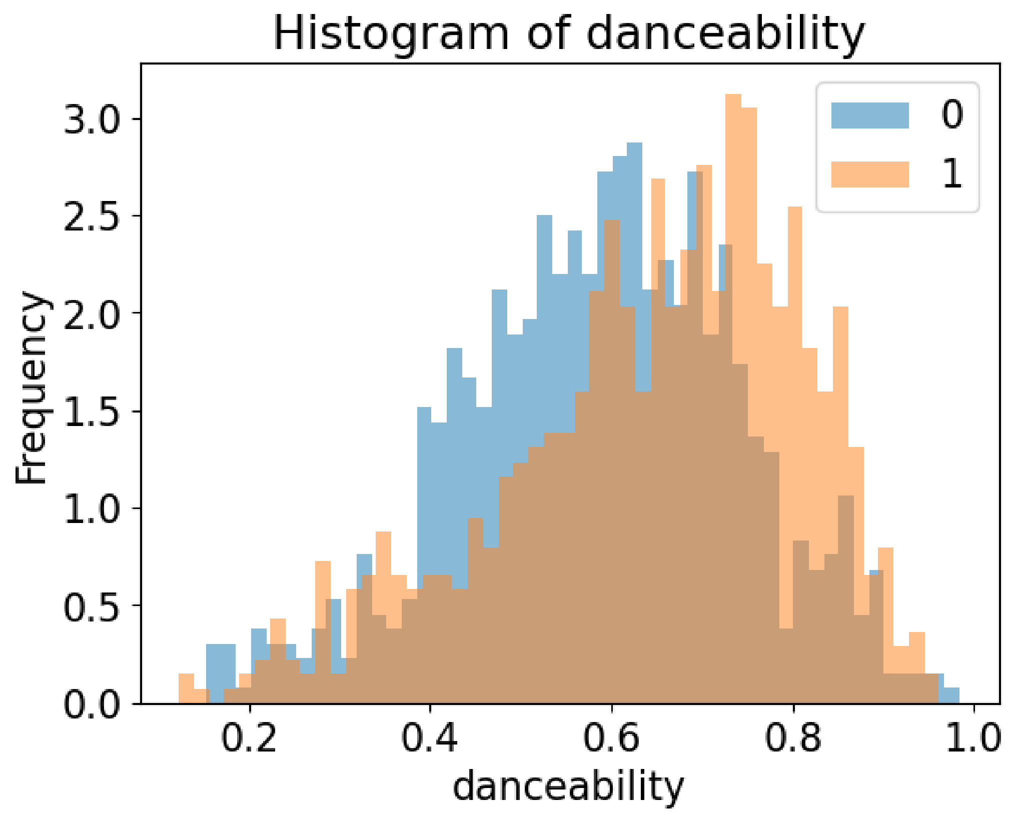
A) Identifying if an email is spam or not
B) Predicting tomorrow's temperature in degrees
C) Classifying different species of flowers
D) Identifying fraudulent credit card transactions

Introduction to Machine Learning

Which of the following is an example of a **regression** problem?

A) Identifying if an email is spam or not
B) Predicting tomorrow’s temperature in degrees
C) Classifying different species of flowers
D) Identifying fraudulent credit card transactions

EDA

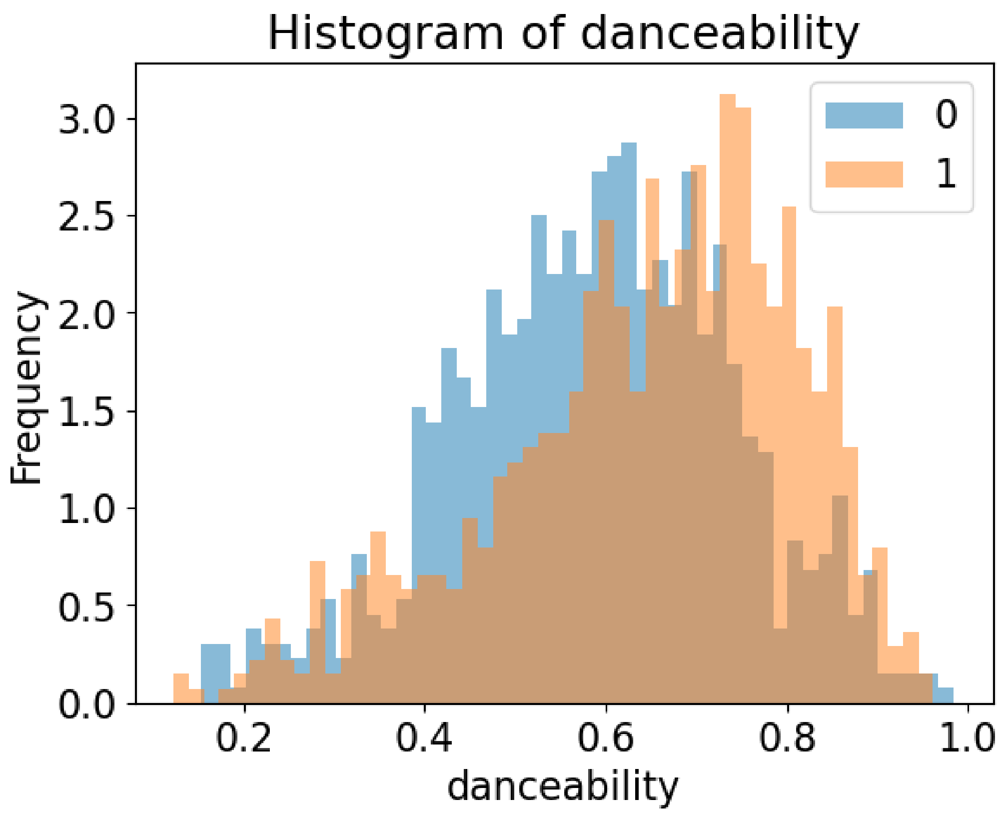


0 = not liked song; 1 = liked song

Based on this histogram, danceability is not a very good discriminant to separate the two classes. Should I remove it from my model?

A. Yes, because it is not informative
B. Yes, because its range is too narrow
C. No, because the classes shows different frequencies across the range
D. No, because it could be informative when combined with other features

EDA



0 = not liked song; 1 = liked song

Based on this histogram, danceability is not a very good discriminant to separate the two classes. Should I remove it from my model?

A. Yes, because it is not informative
B. Yes, because its range is too narrow
C. No, because the classes shows different frequencies across the range
D. No, because it could be informative when combined with other features

Decision trees

Increasing the depth of a decision tree will do all of the following, except one:

A) Improve model performance on training samples
B) Increase the risk of overfitting
C) Increase model complexity
D) Make the model more interpretable

Decision trees

Increasing the depth of a decision tree will do all of the following, except one:

A) Improve model performance on training samples
B) Increase the risk of overfitting
C) Increase model complexity
D) Make the model more interpretable

K-Nearest Neighbors

What is a key **limitation** of k-Nearest Neighbors (kNN)?

A) It requires labeled data
B) It does not work for regression tasks
C) It is computationally expensive for large datasets
D) It cannot handle numerical features

K-Nearest Neighbors

What is a key **limitation** of k-Nearest Neighbors (kNN)?

A) It requires labeled data
B) It does not work for regression tasks
C) It is computationally expensive for large datasets
D) It cannot handle numerical features

K-Nearest Neighbors

All of the following will make a kNN classifier slower at generating predictions, but which one will have the smaller impact?

A. A higher number of features
B. A higher number of training samples
C. A higher value of k
D. Using a distance metric that is computationally expensive (e.g., Mahalanobis distance instead of Euclidean)

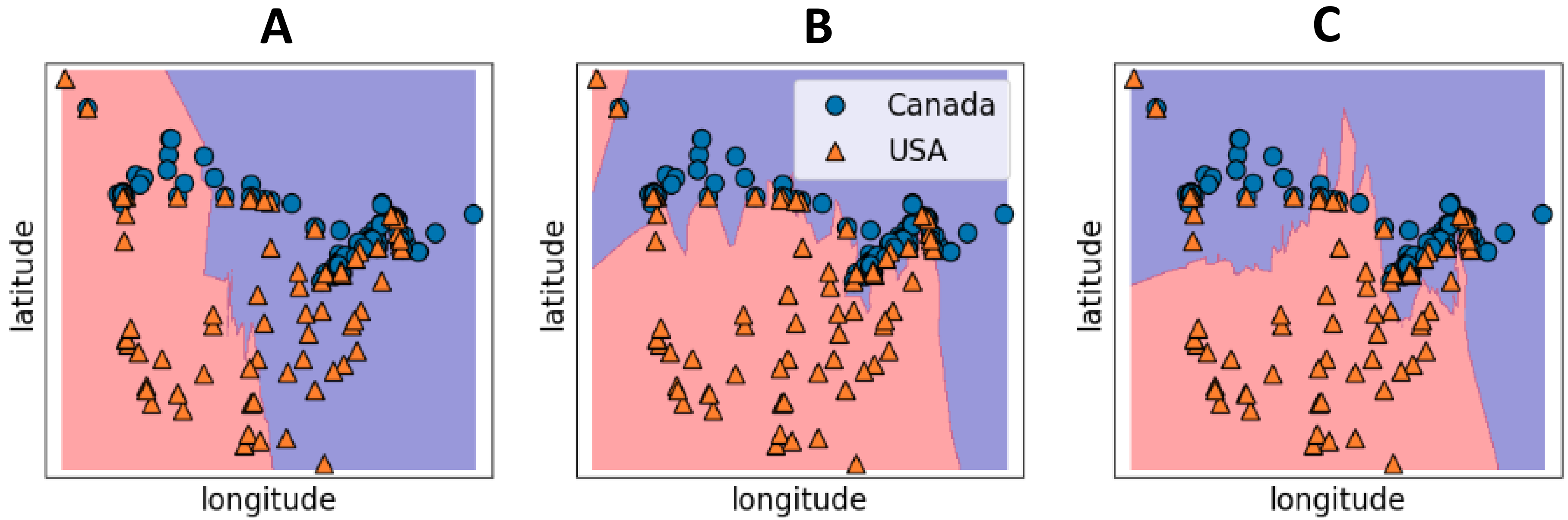
K-Nearest Neighbors

All of the following will make a kNN classifier slower at generating predictions, but which one will have the smaller impact?

A. A higher number of features
B. A higher number of training samples
C. A higher value of k
D. Using a distance metric that is computationally expensive (e.g., Mahalanobis distance instead of Euclidean)

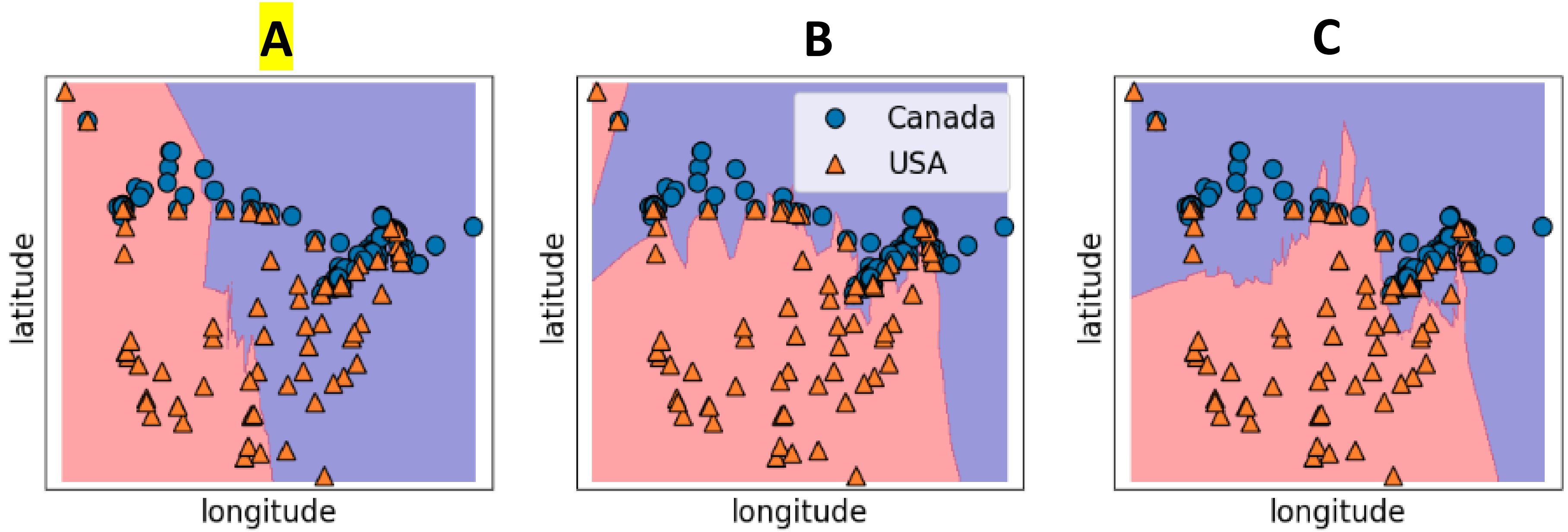
Decision boundaries

The following decision boundaries correspond to kNN classifiers trained with different values of k. Which one do you think was trained with the **highest** value of k?



Decision boundaries

The following decision boundaries correspond to kNN classifiers trained with different values of k. Which one do you think was trained with the **highest** value of k?



Underfitting/overfitting

What should I do to help prevent **overfitting**?

A) Increase the number of features
B) Reduce the amount of training data
C) Use regularization techniques, like Ridge
D) Train for a longer time

Underfitting/overfitting

What should I do to help prevent **overfitting**?

A) Increase the number of features
B) Reduce the amount of training data
C) Use regularization techniques, like Ridge
D) Train for a longer time

Underfitting/overfitting

Which of the following scenarios suggests a model is suffering from **high bias**?

A) The training and test errors are both high and similar in magnitude
B) The training error is low, but the test error is significantly higher
C) The model performs well on the training set but struggles with new data
D) The model's performance improves significantly when adding more features

Underfitting/overfitting

Which of the following scenarios suggests a model is suffering from **high bias**?

A) The training and test errors are both high and similar in magnitude
B) The training error is low, but the test error is significantly higher
C) The model performs well on the training set but struggles with new data
D) The model's performance improves significantly when adding more features

Cross validation

Cross-validation helps by:

A) Increasing dataset size
B) Reducing bias in performance estimates
C) Making training faster
D) Avoiding the need for feature scaling

Cross validation

Cross-validation helps by:

A) Increasing dataset size
B) Reducing bias in performance estimates
C) Making training faster
D) Avoiding the need for feature scaling

Preprocessing

Which of the following is **not** a common preprocessing step?

A) Feature scaling
B) Removing duplicate labels
C) Replacing missing values
D) Converting categorical variables to numerical

Preprocessing

Which of the following is **not** a common preprocessing step?

A) Feature scaling
B) Removing duplicate labels
C) Replacing missing values
D) Converting categorical variables to numerical

Preprocessing

Sophia is a data scientist working on a **sentiment analysis model** for customer reviews. She decides to use **CountVectorizer** from scikit-learn to convert text into numerical features.

After applying **CountVectorizer** to her dataset, she notices something odd:

- The feature matrix has **many columns**, making it very sparse.
- Common words like **"the," "and," "is"** appear frequently, inflating the feature counts.
- Words like **"awesome" and "terrible"**, which are important for sentiment analysis, are **overshadowed** by common words.

Her colleague suggests tweaking **CountVectorizer's parameters** to improve the feature representation. Which of the following would be the **best approach**?

A) Set stop_words='english' to remove common words that don’t add meaning to the sentiment.
B) Set max_features=10 to drastically reduce the vocabulary size.
C) Use binary=True so that word frequency is ignored completely.
D) Remove **low-frequency words** by setting min_df=10 to filter out rare words.

Preprocessing

Sophia is a data scientist working on a **sentiment analysis model** for customer reviews. She decides to use **CountVectorizer** from scikit-learn to convert text into numerical features.

After applying **CountVectorizer** to her dataset, she notices something odd:

- The feature matrix has **many columns**, making it very sparse.
- Common words like **"the," "and," "is"** appear frequently, inflating the feature counts.
- Words like **"awesome" and "terrible"**, which are important for sentiment analysis, are **overshadowed** by common words.

Her colleague suggests tweaking **CountVectorizer's parameters** to improve the feature representation. Which of the following would be the **best approach**?

A) Set stop_words='english' to remove common words that don’t add meaning to the sentiment.
B) Set max_features=10 to drastically reduce the vocabulary size.
C) Use binary=True so that word frequency is ignored completely.
D) Remove **low-frequency words** by setting min_df=10 to filter out rare words.

Hyperparameters tuning

Given an SVM with an **RBF kernel**, increasing the gamma parameter will likely:

A) Make the decision boundary more linear
B) Reduce model complexity
C) Make the model more sensitive to individual data points
D) Decrease the risk of overfitting

Hyperparameters tuning

Given an SVM with an **RBF kernel**, increasing the gamma parameter will likely:

A) Make the decision boundary more linear
B) Reduce model complexity
C) Make the model more sensitive to individual data points
D) Decrease the risk of overfitting

Hyperparameters tuning

Compared to **Grid Search**, what is a key advantage of **Randomized Search**?

A) It guarantees finding the best hyperparameters
B) It reduces computational cost by sampling fewer hyperparameter combinations
C) It always improves model accuracy
D) It is also applicable to regression problems

Hyperparameters tuning

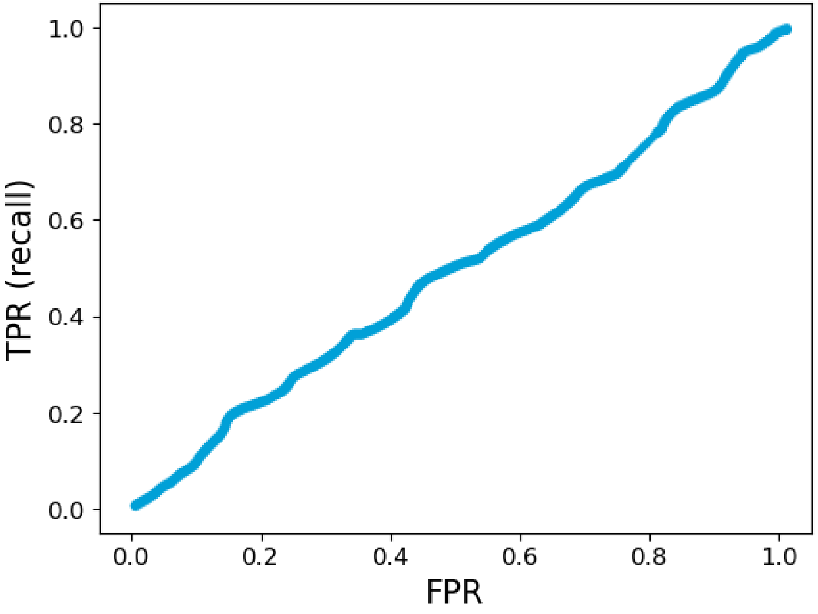
Compared to **Grid Search**, what is a key advantage of **Randomized Search**?

A) It guarantees finding the best hyperparameters
B) It reduces computational cost by sampling fewer hyperparameter combinations
C) It always improves model accuracy
D) It is also applicable to regression problems

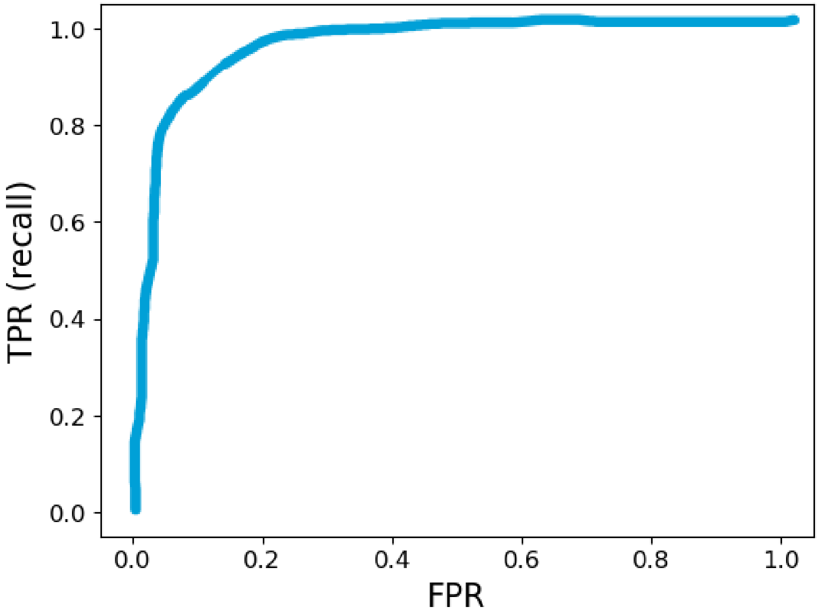
Classification metrics

I am tasked to solve a binary classification problem, but I am lazy and I decide to use a coin toss to assign each sample to a class (head = positive, tail = negative). The classes in the dataset are balanced. What ROC curve better corresponds to my approach?

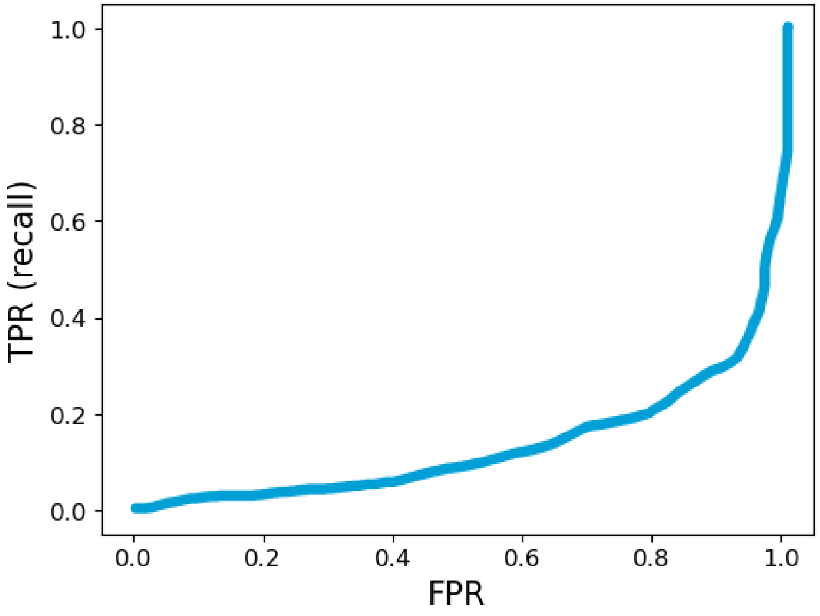
A



B



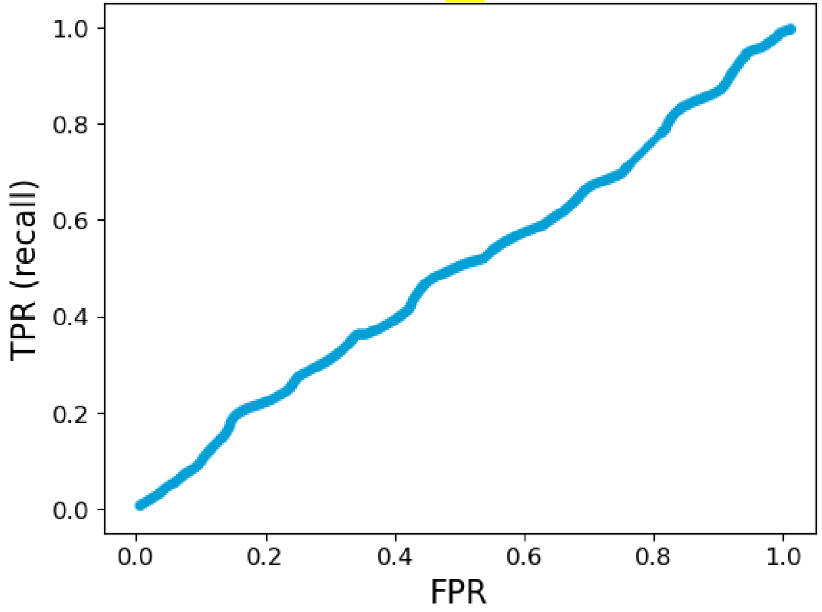
C



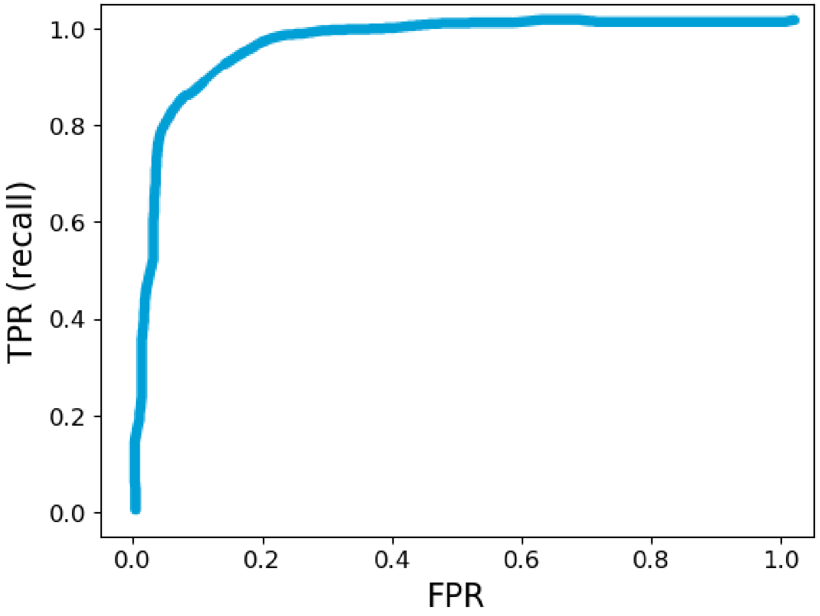
Classification metrics

I am tasked to solve a binary classification problem, but I am lazy and I decide to use a coin toss to assign each sample to a class (head = positive, tail = negative). The classes in the dataset are balanced. What ROC curve better corresponds to my approach?

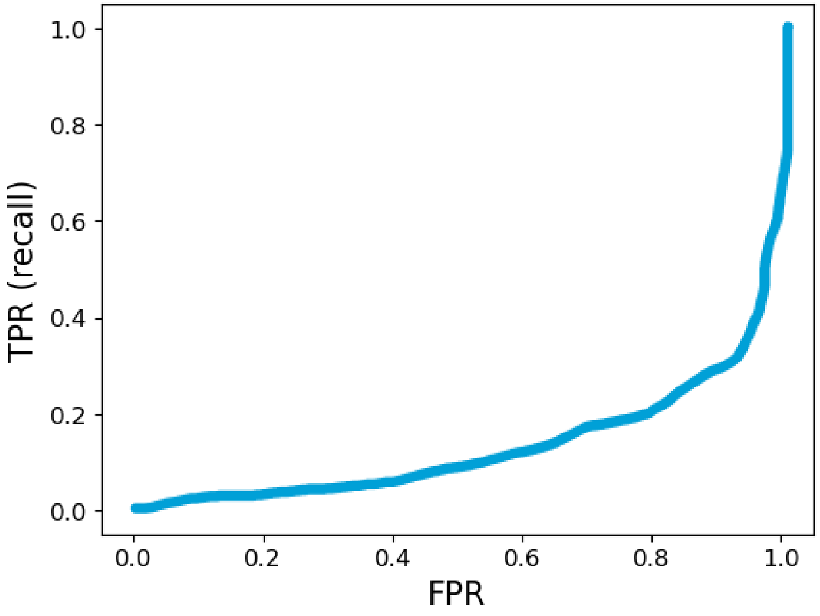
A



B



C



Classification metrics

If a classification model achieves **100% recall**, what can we conclude?

A) The model also has 100% accuracy
B) The model correctly identified all positive samples but may have false positives
C) The model does not make false positive predictions
D) The model has a high precision score

Classification metrics

If a classification model achieves **100% recall**, what can we conclude?

A) The model also has 100% accuracy
B) The model correctly identified all positive samples but may have false positives
C) The model does not make false positive predictions
D) The model has a high precision score

Regression metrics

	fit_time	score_time	test_score	train_score
0	0.002385	0.000832	-0.003547	0.0
1	0.001790	0.000803	-0.001266	0.0
2	0.001433	0.000520	-0.011767	0.0
3	0.002221	0.000332	-0.006744	0.0
4	0.001894	0.000433	-0.076533	0.0
5	0.004854	0.001406	-0.003133	0.0
6	0.002746	0.001011	-0.000397	0.0
7	0.004143	0.001566	-0.003785	0.0
8	0.000652	0.000221	-0.001740	0.0
9	0.000713	0.000226	-0.000117	0.0

The table on the left shows the cross-validation results for a regression problem. Which regressor is likely being used here?

A. DummyRegressor(strategy=”median”)
B. SVR(kernel='rbf')
C. DummyRegressor(strategy=”mean”)
D. SVR(kernel=‘linear')

Regression metrics

	fit_time	score_time	test_score	train_score
0	0.002385	0.000832	-0.003547	0.0
1	0.001790	0.000803	-0.001266	0.0
2	0.001433	0.000520	-0.011767	0.0
3	0.002221	0.000332	-0.006744	0.0
4	0.001894	0.000433	-0.076533	0.0
5	0.004854	0.001406	-0.003133	0.0
6	0.002746	0.001011	-0.000397	0.0
7	0.004143	0.001566	-0.003785	0.0
8	0.000652	0.000221	-0.001740	0.0
9	0.000713	0.000226	-0.000117	0.0

The table on the left shows the cross-validation results for a regression problem. Which regressor is likely being used here?

A. DummyRegressor(strategy=”median”)
B. SVR(kernel='rbf')
C. DummyRegressor(strategy=”mean”)
D. SVR(kernel=‘linear')

Regression metrics

Liam is a financial analyst at a startup that predicts **monthly revenue** for different business units. He is evaluating the model's performance and has to choose between using **Mean Absolute Percentage Error (MAPE)** and **R^2 (coefficient of determination)**.

Liam notices something interesting:

- The model performs **well** for high-revenue business units but **poorly** for smaller ones.
- The **R^2 score is high (0.92)**, but the **MAPE is 40%**, meaning predictions are off by an average of 40% of actual revenue.
- Some business units have **low actual revenue**.

Which metric should Liam trust more in this case, and why?

A) MAPE is better because it considers percentage errors, making it fair across different revenue sizes.
B) R^2 is better because a high value (0.92) means the model explains most of the variance; Liam should use R^2 and ignore MAPE.
C) R^2 is always the best metric for regression, regardless of data characteristics.
D) MAPE may be misleading when actual values are small; Liam should pick R^2 because it gives a more reliable picture of the model's ability to explain variance.

Regression metrics

Liam is a financial analyst at a startup that predicts **monthly revenue** for different business units. He is evaluating the model's performance and has to choose between using **Mean Absolute Percentage Error (MAPE)** and **R^2 (coefficient of determination)**.

Liam notices something interesting:

- The model performs **well** for high-revenue business units but **poorly** for smaller ones.
- The **R^2 score is high (0.92)**, but the **MAPE is 40%**, meaning predictions are off by an average of 40% of actual revenue.
- Some business units have **low actual revenue**.

Which metric should Liam trust more in this case, and why?

A) MAPE is better because it considers percentage errors, making it fair across different revenue sizes.
B) R^2 is better because a high value (0.92) means the model explains most of the variance; Liam should use R^2 and ignore MAPE.
C) R^2 is always the best metric for regression, regardless of data characteristics.
D) MAPE may be misleading when actual values are small; Liam should pick R^2 because it gives a more reliable picture of the model's ability to explain variance.

GL on Midterm 1! :)