

CS/ENGR M148 L8: Classification

Sandra Batista

Quiz on Problem Set 1 today!

Only 15 minutes. Multi-select, multiple choice, T/F questions.

Please bring laptop to take quiz and hard copy of notes.

For CAE accommodations:

- 1) Course accommodation for this quiz only
- 2) Schedule your testing at CAE testing center for quizzes, midterm (100 minutes regular time), and final² by 10/29/24.

Please contact TA first for homework submission help. I can help if TA cannot resolve.

This week in discussion section:
Lab on logistic regression

Project Data Check-in: Your team will need to demonstrate a logistic regression model on your project data.

New for Project Check-ins Early:

11am-11:50am Fridays in Boelter 5436 with our wonderful
TA Yihe

Join our slido for the week...

<https://app.sli.do/event/6S6WBqZX9qAfDq4tn2QDoZ>



Today's Learning Objectives

Students will be able to:

- Review: Understand **classification problems** and **use logistic regression** on real data
- Review: Evaluate classification problems with quantitative metrics
- Apply KNN algorithm by hand to a small sample data set

Categorical Variables

Categorical variables are variables that do not have numerical measurements (e.g. neighborhood in Ames housing data)

Categorical variables can be **ordinal** if categories can be sorted.

Categorical variables can be **nominal** if categories do not have specific order.

Categorical variables can be made converted to numeric values (e.g. one-hot encoding)

Classification

A binary response is often referred to as the **class label** of the observation.

Classification problems: Prediction problems with binary responses that involve *classifying* each observation as belonging to one of the two classes.

(It is possible to have more than 2 classes...)

Classification

A binary response is often referred to as the **class label** of the observation.

Classification problems: Prediction problems with binary responses that involve *classifying* each observation as belonging to one of the two classes.

(It is possible to have more than 2 classes...)

Logistic regression

As an example we'll consider UCI shopping data set in notebook, let's predict purchase made or not.

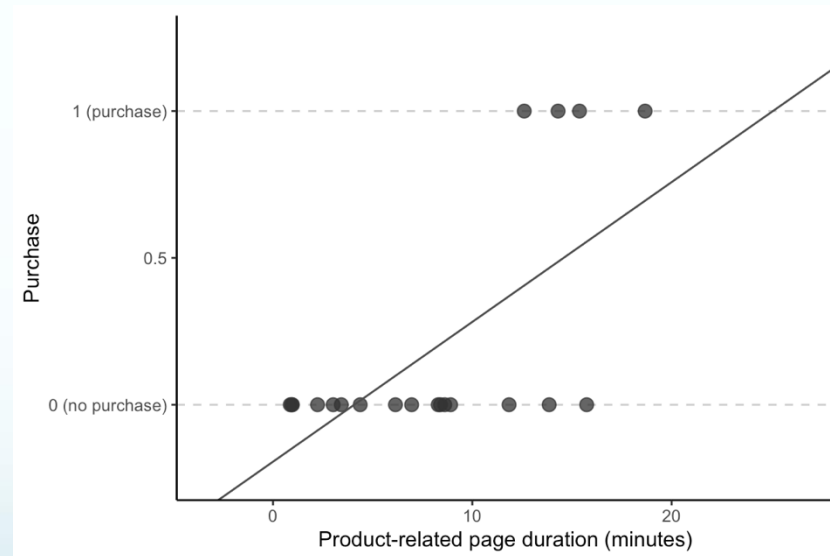
With binary response variable linear combinations of variable and least squares won't work:

$$\text{predicted purchase} = -0.194 + 0.048 \times \text{product-related duration},$$

Logistic regression

As an example we'll consider UCI shopping data set in notebook, let's predict purchase made or not.

With binary response variable linear combinations of variable and least squares won't work:



Logistic regression

Rather than trying to predict binary response variable, we try to predict continuous **probability of binary variable...**

predicted purchase *probability* = $b_0 + b_1 \times$ product-related duration.

But something is still wrong, what?

Logistic regression

We apply a **logistic** transformation to the equation to get valid probabilities from the predictor

Logistic regression uses a *logistic* linear combination to predict the *probability* of a class label (success).

Odds Ratio

The odds (odds ratio) corresponds to the probability of a "success," p , divided by the probability of a "failure," $1 - p$:

$$\frac{p}{1 - p}.$$

The odds ratio is bounded between 0 and ∞ .

Log Odds or Logit Function

The log odds (logit function) corresponds to the logarithm of the odds ratio:

$$\log \left(\frac{p}{1-p} \right).$$

The log odds is an unbounded continuous number.

We apply the logit function to the probability, so it equals a linear combination of predictors:

$$\log \left(\frac{p}{1-p} \right) = b_0 + b_1 \times \text{product-related duration}$$

Logistic Function

*We invert the logit function and solve for the probability to get the **logistic function**:*

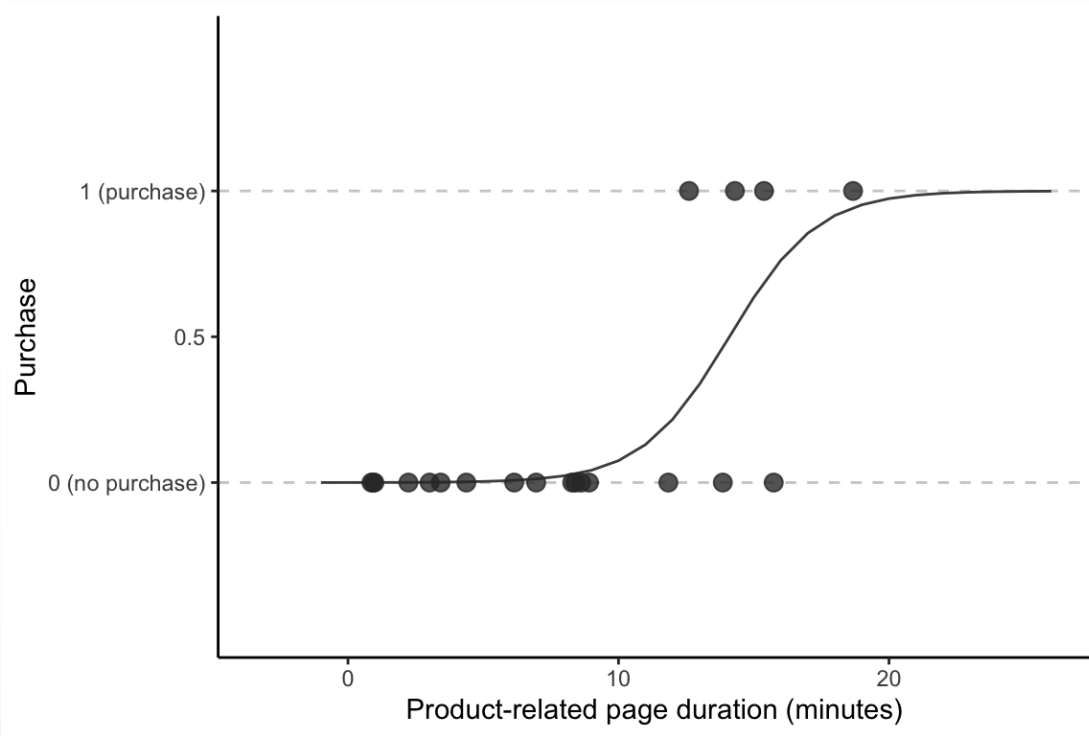
Logistic regression computes binary response probability predictions, p , based on the logistic-transformed linear combination:

$$p = \frac{1}{1 + e^{-(b_0 + b_1 x)}},$$

where x is a relevant predictive feature.

Logistic regression uses this function to compute values for Parameters.

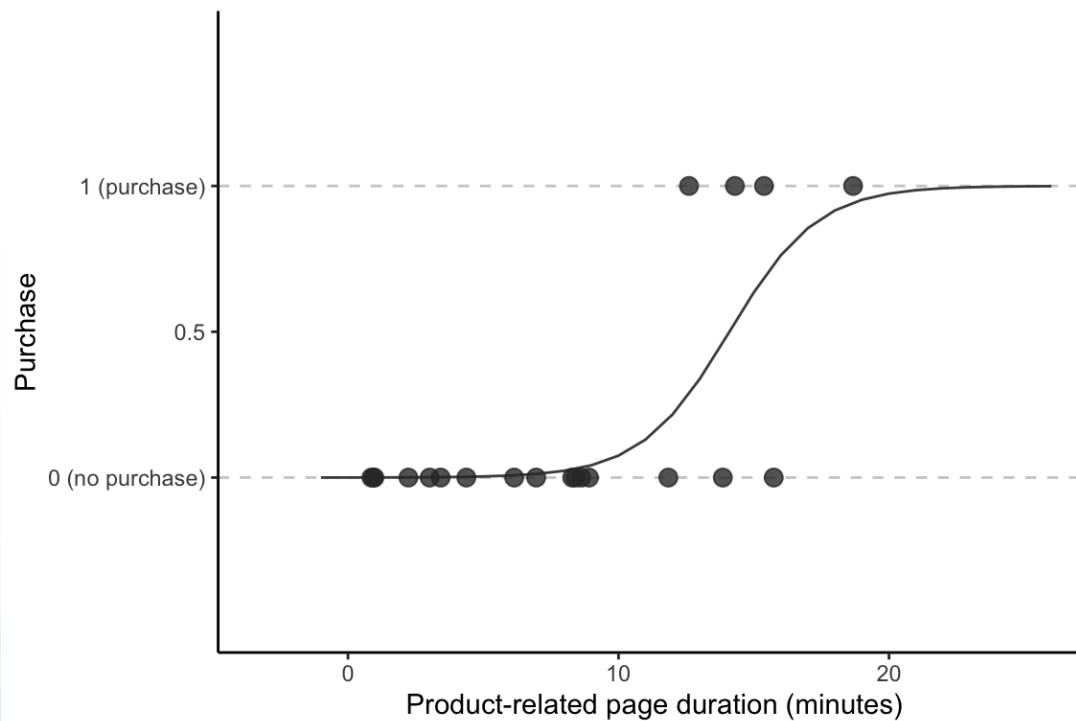
Logistic Function



$$\text{predicted purchase probability} = \frac{1}{1 + e^{-(-8.639 + 0.613 \text{product-related duration})}}$$

How to get predictions?

*If p greater than or equal to threshold (.5) predict 1 (or purchase)
Otherwise 0 (no purchase)*

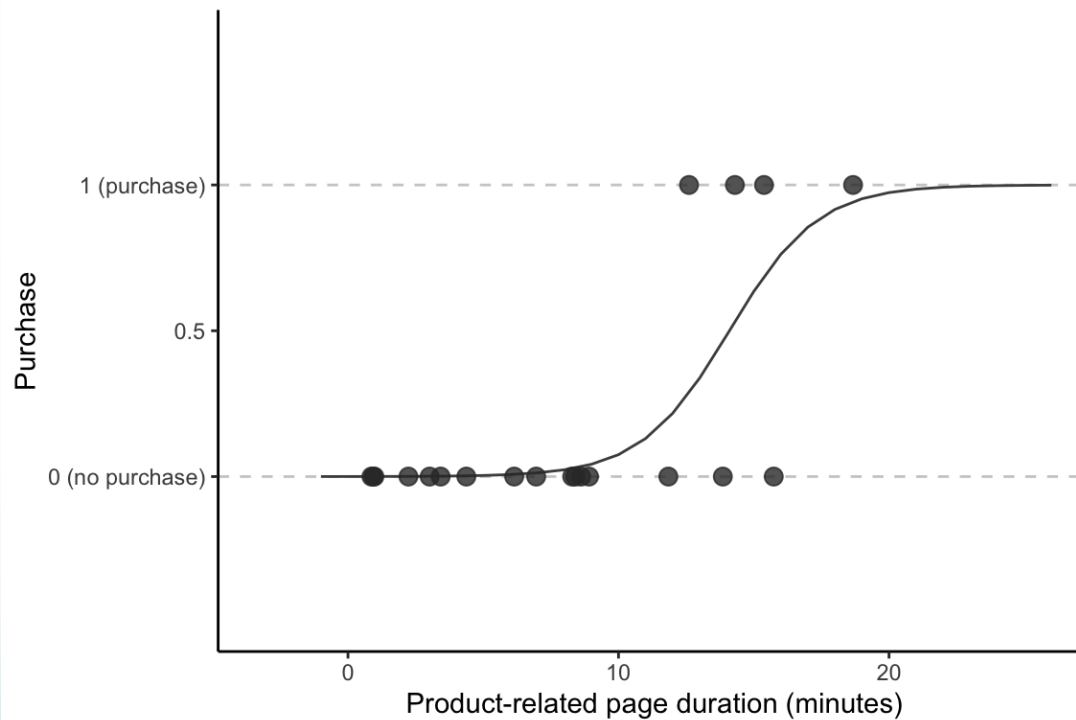


$$\text{predicted purchase probability} = \frac{1}{1 + e^{-(-8.639 + 0.613 \text{product-related duration})}}$$

How to choose threshold?

Consider **class imbalance** in training data set

Practice: set threshold to proportion of positive class labels in training data



$$\text{predicted purchase probability} = \frac{1}{1 + e^{-(-8.639 + 0.613 \text{product-related duration})}}$$

What is the loss function?

***Logistic Loss function** to minimize, log likelihood function*

$$\sum_{i \text{ in pos class}} (-\log p_i) + \sum_{i \text{ in neg class}} (-\log(1 - p_i)).$$

No nice closed form. Can use techniques such as Maximum Likelihood Estimation (MLE)

Today's Learning Objectives

Students will be able to:

- ✓ Review: Understand **classification problems** and **use logistic regression** on real data
- ✗ Review: Evaluate classification problems with quantitative metrics
- ✗ Apply KNN algorithm by hand to a small sample data set

Evaluating Classification

Prediction accuracy is proportion of observations for which the binary predicted response label matches the observed response label.





$$\text{prediction accuracy} = \frac{(\text{number of correct predictions})}{n}$$

Prediction error corresponds to the proportion of observations for which the binary predicted response label is *different* from the observed response label.

$$\text{prediction error} = \frac{(\text{number of incorrect predictions})}{n}$$

Confusion matrix

The **confusion matrix** is a 2-by-2 table that cross-tabulates the predicted and observed binary response.

		Predicted	
		positive	negative
Observed	positive	 # true pos	 # false neg
	negative	 # false pos	 # true neg

Sensitivity or True Positive Rate

The **true positive rate** (often called “**sensitivity**” or “**recall**”) is the proportion of positive class observations whose class is correctly predicted.

$$\begin{aligned}\text{true positive rate} &= \frac{(\text{number of correctly predicted positive class obs})}{(\text{number of positive class observations})} \\ &= \frac{(\text{number of true positives})}{(\text{number of positive class observations})}\end{aligned}$$

Decreasing threshold for probability **increases sensitivity** but decreases specificity

Specificity or True Negative Rate

The **true negative rate** (often called “**specificity**”) is the proportion of negative class observations whose class is correctly predicted

The **false positive rate** is the proportion of negative observations *incorrectly* predicted to be positive.

Increasing threshold for probability **increases specificity** but decreases sensitivity

Tradeoff between sensitivity and specificity

Your turn

What is the confusion matrix for this data?

What is the specificity or true negative rate?

What is the sensitivity or true positive rate?

What is accuracy and error?

True	Predicted	True	Predicted
------	-----------	------	-----------

1	0.34 (0)
---	----------

1	0.33 (0)
---	----------

1	0.62 (1)
---	----------

1	0.43 (0)
---	----------

1	0.42 (0)
---	----------

0	0.08 (0)
---	----------

0	0.35 (0)
---	----------

0	0.06 (0)
---	----------

0	0.05 (0)
---	----------

0	0.08 (0)
---	----------

0	0.06 (0)
---	----------

0	0.19 (0)
---	----------

0	0.24 (0)
---	----------

0	0.09 (0)
---	----------

0	0.13 (0)
---	----------

0	0.43 (0)
---	----------

0	0.38 (0)
---	----------

0	0.32 (0)
---	----------

0	0.5 (1)
---	---------

0	0.39 (0)
---	----------

Your turn

What is the confusion matrix for this data?

What is the specificity or true negative rate?

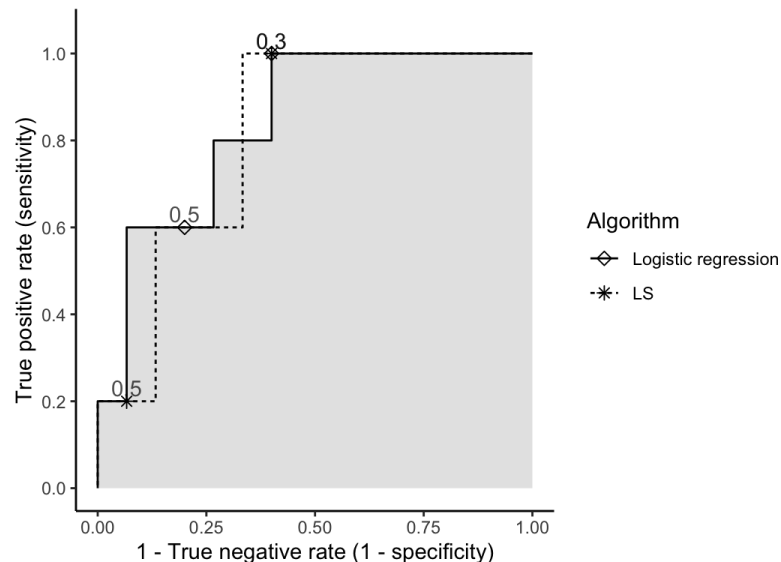
What is the sensitivity or true positive rate?

What is accuracy and error?

ROC Curves

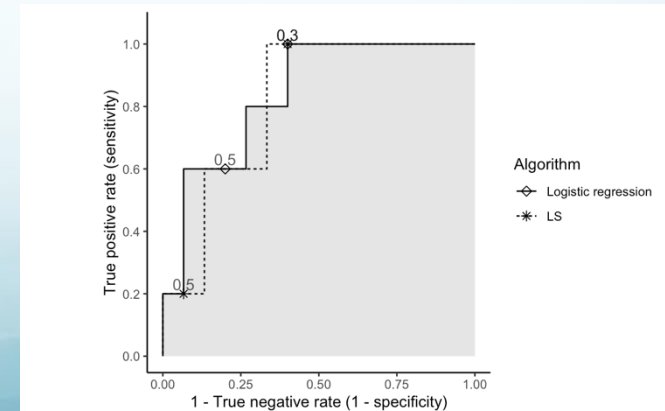
Receiver Operating Characteristics (ROC) curve plot true positive rate against true negative rate for various thresholds to compare models and algorithms.

Area under the curve (AUC) quantifies predictive potential of algorithm by computing the literal area under the ROC curve.



How to create ROC Curves

1. For each possible threshold, run the classification algorithm on the training (or validation) data set (or even cross-validation data sets)
2. Calculate the true positive rate and true negative rate
3. Plot the true negative rate against the true positive rate for each threshold
4. Calculate the area under the ROC curve for AUC (perfect classification is 1, but .8 or better is good)
5. Often use CV on validation set to calculate ROC and compare across algorithms



Your turn:

Logistic Regression

Please get the Jupyter notebook for logistic regression on shopping data:

Go to:

<https://colab.research.google.com/drive/1wazvX6RQGUYRMJEK46tRgHqMyJokTdFC?usp=sharing>

Save a copy to your Google Drive and keep notes there...

Today's Learning Objectives

Students will be able to:

- ✓ Review: Understand **classification problems** and **use logistic regression** on real data
- ✓ Review: Evaluate classification problems with quantitative metrics
- ✗ Apply KNN algorithm by hand to a small sample data set

Similarity-Based Learning

Similarity-based learning is classifying new data based on previous observations.

One of the simplest and best known machine learning algorithms for this type of reasoning is called the **nearest neighbor** algorithm.

Motivational example: An alien sees an animal that we know is a platypus, but the alien has not seen before.

*The alien has seen and hear ducks, frogs, and lions.
How will this alien classify this new animal?*






	Grrrh!			Score
	✓	X	X	1
	X	✓	X	1
	X	✓	✓	2

Figure: Matching animals you remember to the features of the unknown animal described by the sailor. Note: The images used in this figure were created by Jan Gillbank for the English for the Australian Curriculum website used under the Create Commons Attribution 3.0 Unported. The images were sourced via Wikimedia Commons.

Table: *The speed and agility ratings for 20 college athletes labelled with the decisions for whether they were drafted or not.*

<u>ID</u>	<u>Speed</u>	<u>Agility</u>	<u>Draft</u>	<u>ID</u>	<u>Speed</u>	<u>Agility</u>	<u>Draft</u>
1	2.50	6.00	No	11	2.00	2.00	No
2	3.75	8.00	No	12	5.00	2.50	No
3	2.25	5.50	No	13	8.25	8.50	No
4	3.25	8.25	No	14	5.75	8.75	Yes
5	2.75	7.50	No	15	4.75	6.25	Yes
6	4.50	5.00	No	16	5.50	6.75	Yes
7	3.50	5.25	No	17	5.25	9.50	Yes
8	3.00	3.25	No	18	7.00	4.25	Yes
9	4.00	4.00	No	19	7.50	8.00	Yes
10	4.25	3.75	No	20	7.25	5.75	Yes

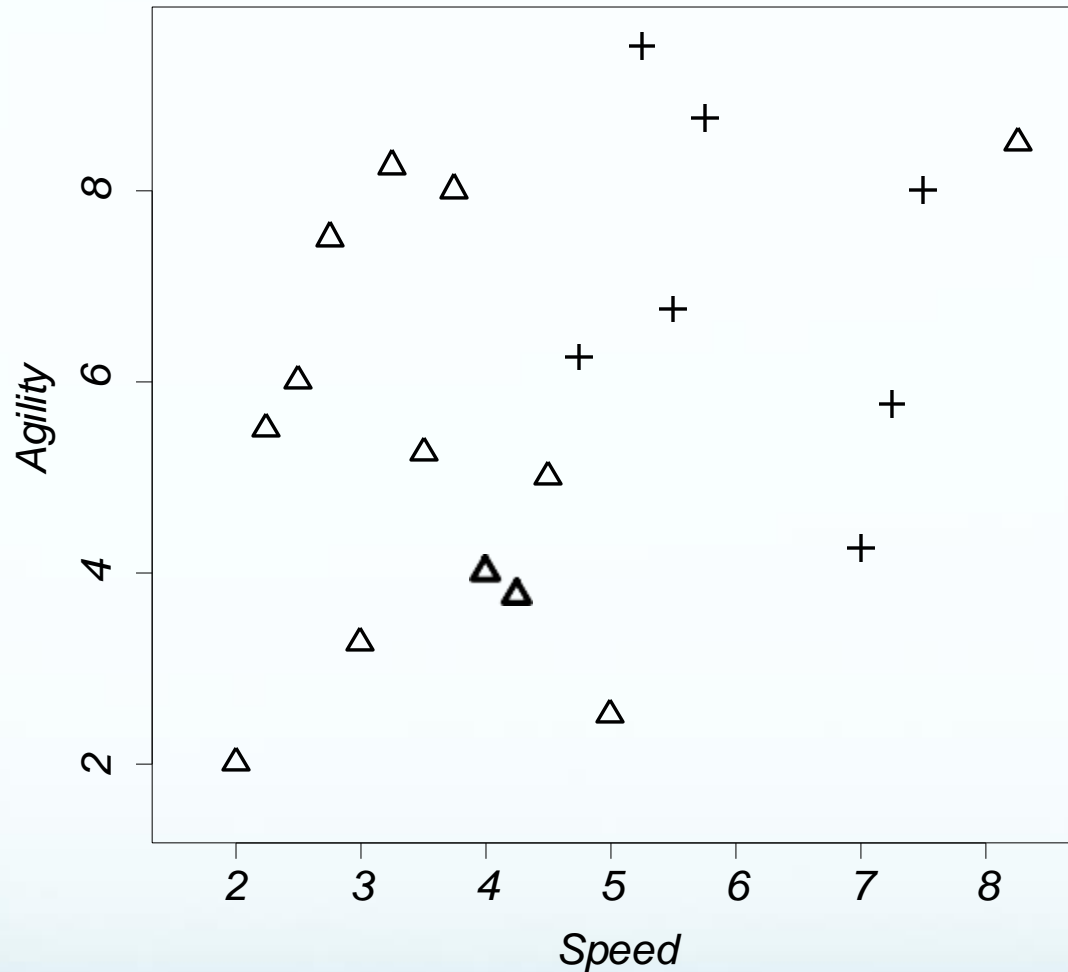


Figure: The speed versus agility for athletes. Triangles represent 'Non-draft' instances and crosses represent the 'Draft' instances.

Similarity Metrics

- A *similarity metric* measures the similarity between two instances according to a feature space
- Mathematically, a *metric* must conform to the following four criteria:
 - 1 *Non-negativity*: $\text{metric}(\mathbf{a}, \mathbf{b}) \geq 0$
 - 2 *Identity*: $\text{metric}(\mathbf{a}, \mathbf{b}) = 0 \iff \mathbf{a} = \mathbf{b}$
 - 3 *Symmetry*: $\text{metric}(\mathbf{a}, \mathbf{b}) = \text{metric}(\mathbf{b}, \mathbf{a})$
 - 4 *Triangular Inequality*:
 $\text{metric}(\mathbf{a}, \mathbf{b}) \leq \text{metric}(\mathbf{a}, \mathbf{c}) + \text{metric}(\mathbf{b}, \mathbf{c})$

where $\text{metric}(\mathbf{a}, \mathbf{b})$ is a function that returns the distance between two instances \mathbf{a} and \mathbf{b} .

Euclidean distance

*One of the best known metrics is **Euclidean distance** which computes the length of the straight line between two points. Euclidean distance between two instances **a** and **b** in a m -dimensional feature space is defined as:*

$$Euclidean(\mathbf{a}, \mathbf{b}) = \sqrt{\sum_{i=1}^m (\mathbf{a}[i] - \mathbf{b}[i])^2}$$

Example

The Euclidean distance between instances d_{12} ($SPEED= 5.00$, $AGILITY= 2.5$) and d_5 ($SPEED= 2.75$, $AGILITY= 7.5$) in Table [2](#)^[25] is:

Manhattan distance

The *Manhattan distance* between two instances **a** and **b** in a feature space with m dimensions is:¹

$$\text{Manhattan}(\mathbf{a}, \mathbf{b}) = \sum_{i=1}^m \text{abs}(\mathbf{a}[i] - \mathbf{b}[i])$$

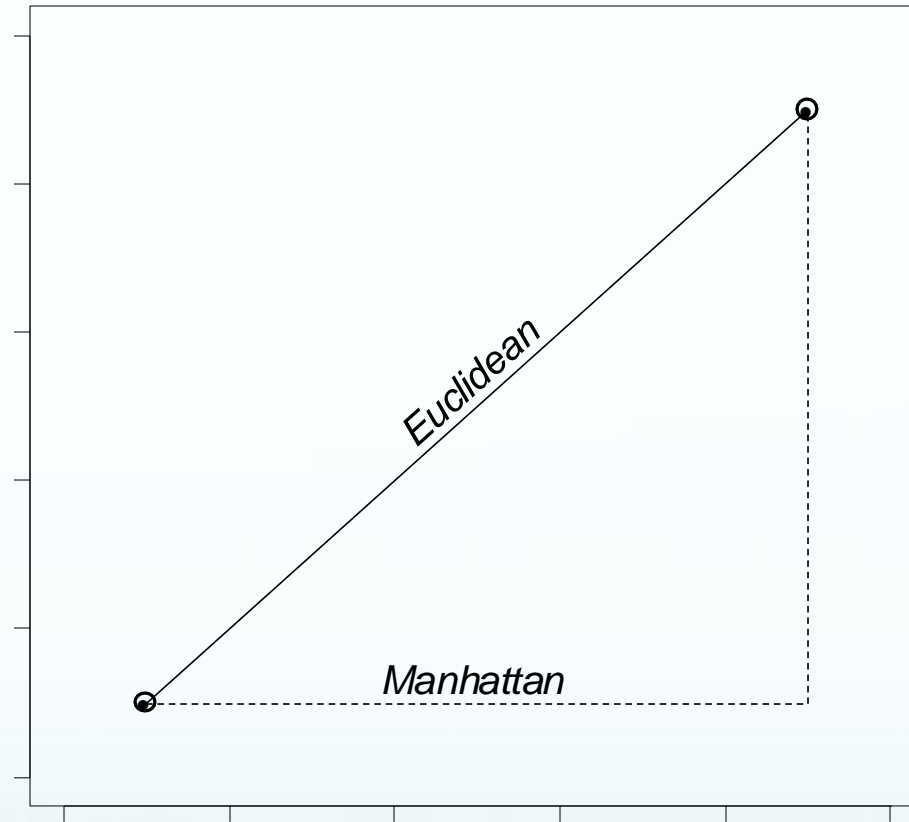


Figure: *The Manhattan and Euclidean distances between two points.*

Example

The Manhattan distance between instances d_{12} ($SPEED= 5.00$, $AGILITY= 2.5$) and d_5 ($SPEED= 2.75$, $AGILITY= 7.5$) in Table [2](#)^[25] is:

Minkowski distance

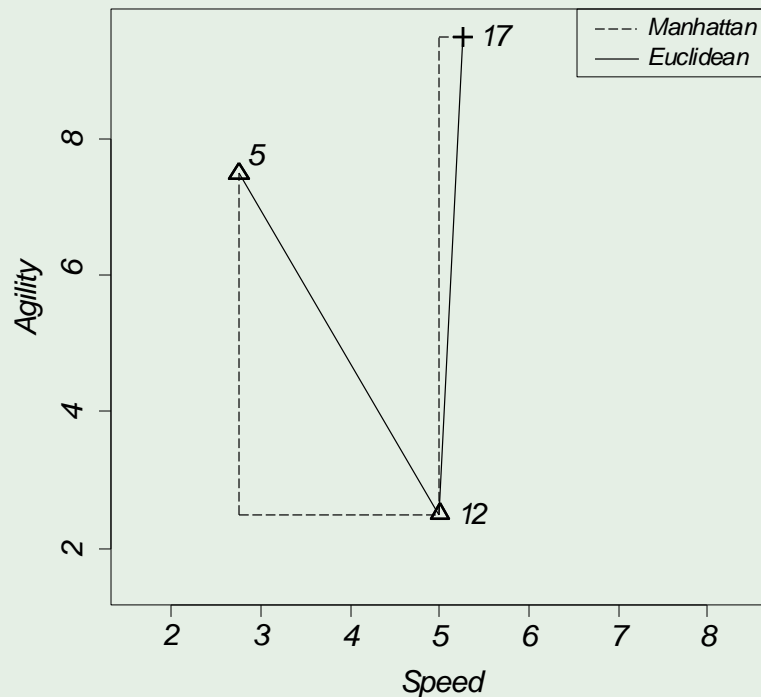
The *Minkowski distance* between two instances **a** and **b** in a feature space with m descriptive features is:

$$\text{Minkowski}(\mathbf{a}, \mathbf{b}) = \left(\sum_{i=1}^m \text{abs}(\mathbf{a}[i] - \mathbf{b}[i])^p \right)^{\frac{1}{p}}$$

where different values of the parameter p result in different distance metrics

- The Minkowski distance with $p = 1$ is the Manhattan distance and with $p = 2$ is the Euclidean distance.
- The larger the value of p the more emphasis is placed on the features with large differences in values

<i>Instance ID</i>	<i>Instance ID</i>	<i>Manhattan (Minkowski $p=1$)</i>	<i>Euclidean (Minkowski $p=2$)</i>
12	5	7.25	5.4829
12	17	7.25	8.25



The Manhattan and Euclidean distances between instances \mathbf{d}_{12} (SPEED= 5.00, AGILITY= 2.5) and \mathbf{d}_5 (SPEED= 2.75, AGILITY= 7.5) and between instances \mathbf{d}_{12} and \mathbf{d}_{17} (SPEED= 5.25, AGILITY= 9.5).

kNN (k nearest neighbor)

1. As in the general problem of classification, we have a set of data points for which we know the correct class labels.
2. When we get a new data point, we compare it to each of our existing data points and find similarity.
3. Take the most similar k data points (k nearest neighbors).
4. From these k data points, take the majority vote of their labels. The winning label is the label/class of the new datapoint.

Choice of k will affect classification and is hyperparameter.

Table: *The speed and agility ratings for 20 college athletes labelled with the decisions for whether they were drafted or not.*

<u>ID</u>	<u>Speed</u>	<u>Agility</u>	<u>Draft</u>	<u>ID</u>	<u>Speed</u>	<u>Agility</u>	<u>Draft</u>
1	2.50	6.00	No	11	2.00	2.00	No
2	3.75	8.00	No	12	5.00	2.50	No
3	2.25	5.50	No	13	8.25	8.50	No
4	3.25	8.25	No	14	5.75	8.75	Yes
5	2.75	7.50	No	15	4.75	6.25	Yes
6	4.50	5.00	No	16	5.50	6.75	Yes
7	3.50	5.25	No	17	5.25	9.50	Yes
8	3.00	3.25	No	18	7.00	4.25	Yes
9	4.00	4.00	No	19	7.50	8.00	Yes
10	4.25	3.75	No	20	7.25	5.75	Yes

Example

- *Should we draft an athlete with the following profile:*

SPEED= 6.75, AGILITY= 3

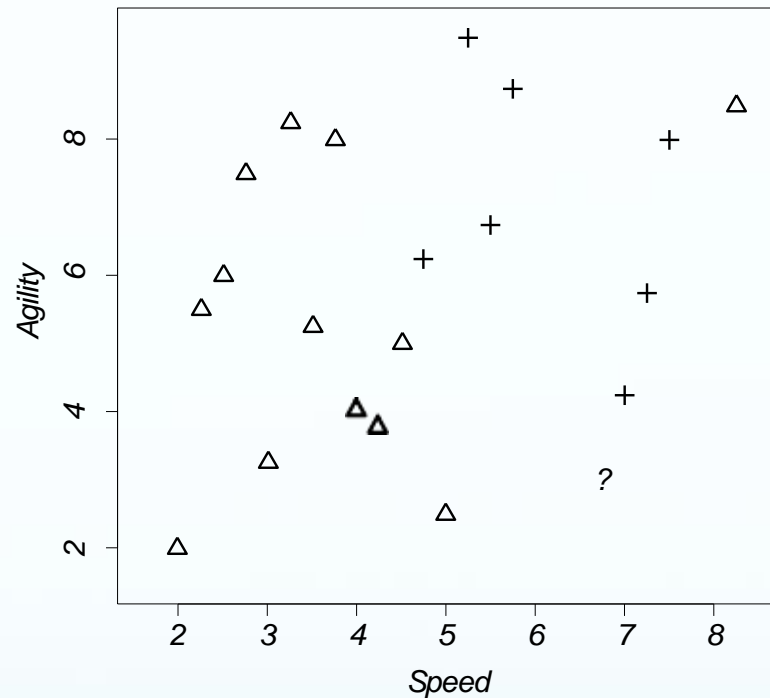


Figure: A plot of speed vs agility for athletes. The triangles represent 'Non-draft' instances and the crosses represent the 'Draft' instances.

Applying KNN algorithm

1. Calculate distance to all other points

Table: The distances (Dist.) between the query instance with *SPEED* = 6.75 and *AGILITY* = 3.00 and each instance in Table 2 ^[25].

<i>ID</i>	<i>SPEED</i>	<i>AGILITY</i>	<i>DRAFT</i>	<i>Dist.</i>	<i>ID</i>	<i>SPEED</i>	<i>AGILITY</i>	<i>DRAFT</i>	<i>Dist.</i>
18	7.00	4.25	yes	1.27	11	2.00	2.00	no	4.85
12	5.00	2.50	no	1.82	19	7.50	8.00	yes	5.06
10	4.25	3.75	no	2.61	3	2.25	5.50	no	5.15
20	7.25	5.75	yes	2.80	1	2.50	6.00	no	5.20
9	4.00	4.00	no	2.93	13	8.25	8.50	no	5.70
6	4.50	5.00	no	3.01	2	3.75	8.00	no	5.83
8	3.00	3.25	no	3.76	14	5.75	8.75	yes	5.84
15	4.75	6.25	yes	3.82	5	2.75	7.50	no	6.02
7	3.50	5.25	no	3.95	4	3.25	8.25	no	6.31
16	5.50	6.75	yes	3.95	17	5.25	9.50	yes	6.67

Applying KNN algorithm

2. Take the k nearest neighbors

3. Take majority vote for new label

Table: The distances (Dist.) between the query instance with $SPEED = 6.75$ and $AGILITY = 3.00$ and each instance in Table 2 [25].

ID	SPEED	AGILITY	DRAFT	Dist.	ID	SPEED	AGILITY	DRAFT	Dist.
18	7.00	4.25	yes	1.27	11	2.00	2.00	no	4.85
12	5.00	2.50	no	1.82	19	7.50	8.00	yes	5.06
10	4.25	3.75	no	2.61	3	2.25	5.50	no	5.15
20	7.25	5.75	yes	2.80	1	2.50	6.00	no	5.20
9	4.00	4.00	no	2.93	13	8.25	8.50	no	5.70
6	4.50	5.00	no	3.01	2	3.75	8.00	no	5.83
8	3.00	3.25	no	3.76	14	5.75	8.75	yes	5.84
15	4.75	6.25	yes	3.82	5	2.75	7.50	no	6.02
7	3.50	5.25	no	3.95	4	3.25	8.25	no	6.31
16	5.50	6.75	yes	3.95	17	5.25	9.50	yes	6.67

What happens for new athlete for $k=1$? $k=3$? For $k=5$?

Decision Boundaries

Decision boundaries are the surfaces separating classes in classification.

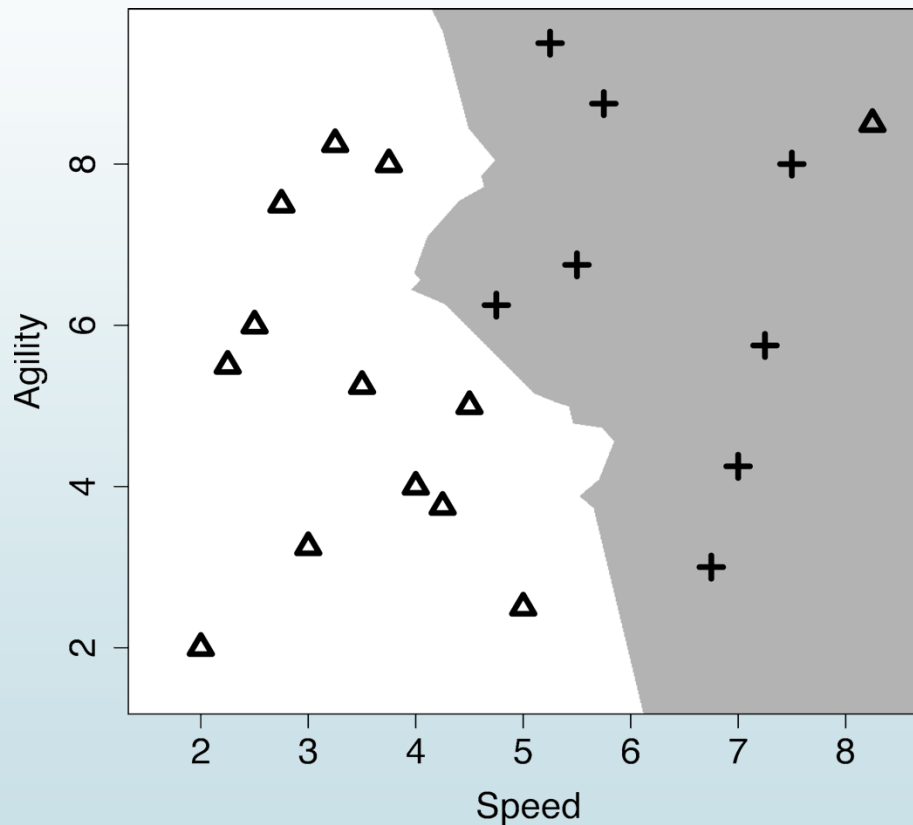


Figure: The decision boundary using majority classification of the nearest 3 neighbors.

Decision Boundaries

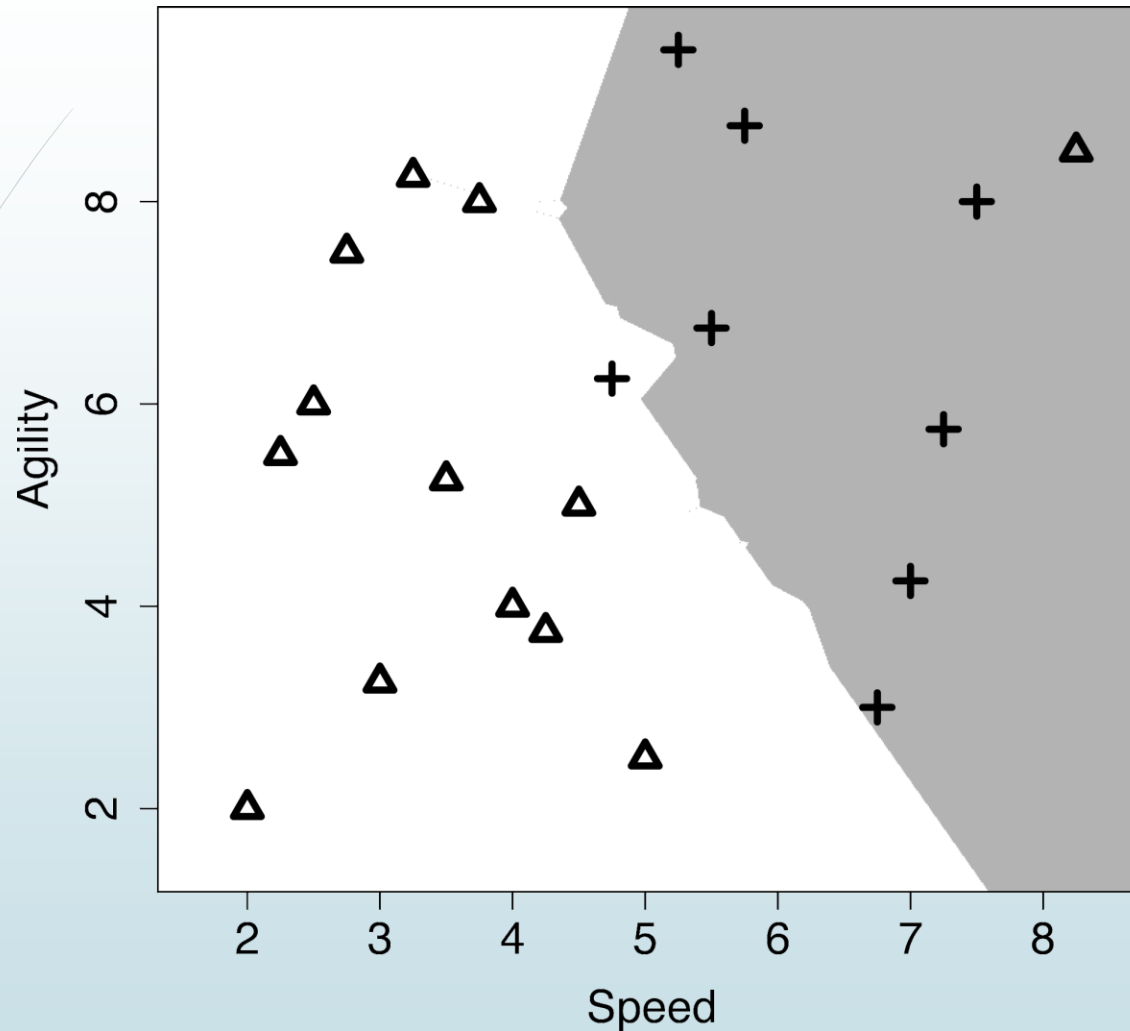


Figure: The decision boundary using majority classification of the nearest 5 neighbors.

[Kelleher et al 2015]

Decision Boundaries

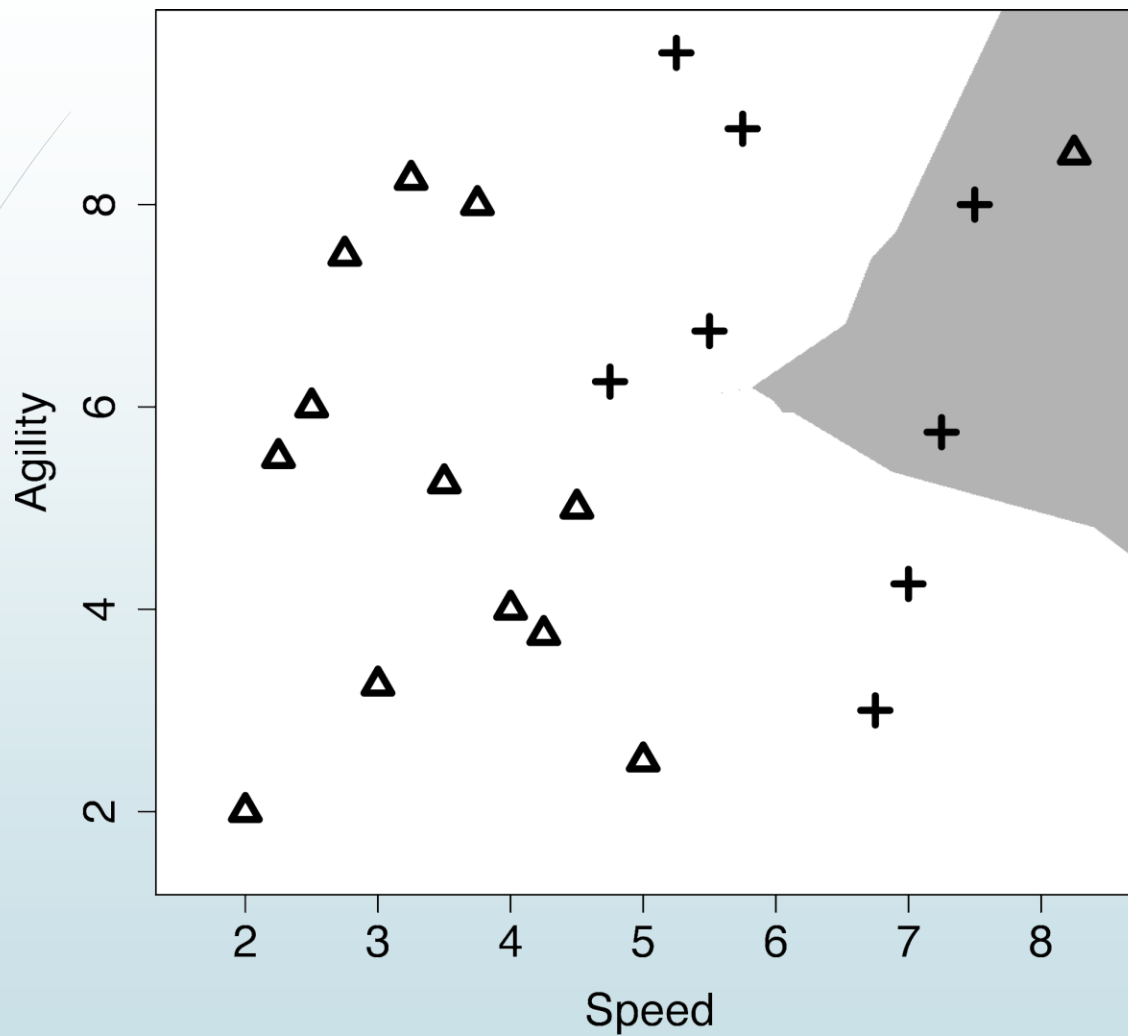


Figure: The decision boundary when k is set to 15.

Today's Learning Objectives

Students will be able to:

- ✓ Review: Understand **classification problems** and **use logistic regression** on real data
- ✓ Review: Evaluate classification problems with quantitative metrics
- ✓ Apply KNN algorithm by hand to a small sample data set

Citations:

Yu, B., & Barter, R. L. (2024). Veridical data science: The practice of responsible data analysis and decision making. The MIT Press.

Shah. C. (2020) A hands-on introduction to data science. Cambridge University Press.

Kelleher, J. D., MacNamee, B., D'Arcy, A. (2015). Fundamentals of Machine Learning for Predictive Data Analytics: Algorithms, Worked Examples, and Case Studies. Cambridge, MA: MIT Press. ISBN: 978-0-262-02944-5