# Machine Learning & JMP

## What Is It, and Do We Do It?

# Spoiler Alert: Yes, we do!

And we always have!

# What is Machine Learning?

- According to Wikipedia:
  "**Machine learning** is the study of computer algorithms that improve automatically through experience."
- ?!?!?
- From Encyclopaedia Britannica:
  "**Machine learning**, in artificial intelligence (a subject within computer science), discipline concerned with the implementation of computer software that can learn autonomously."

# My Definition

- Machine Learning is the current buzz-phrase meant to encompass the computer algorithms used to make decisions, predictions, or classifications based on data.

# Why Machine Learning?

## What's it good for, anyway?

- Categorize people or things
- Predict likely outcomes
- Identify previously unknown patterns or relationships
- Detect anomalous or unexpected behaviors

- All this is done using various algorithms written for different types of tasks

# Types of Machine Learning

## Supervised vs Unsupervised

- Example inputs and outputs provided; algorithm determines relationship(s)
- Decision Trees
- Neural Networks
- Random Forests
- Regression
- Support Vector Machines
- K-Nearest Neighbors
- Naïve Bayes

- Example inputs, but no outputs; algorithm does all the work
- Clustering
- Self-Organizing Maps
- Association Analysis
- Singular Value Decomposition
  - JMP uses SVD as part of several routines – prominently in Text Explorer

# Types of Machine Learning

## Supervised vs Unsupervised

- Example inputs and outputs provided; algorithm determines relationship(s)
- Decision Trees
- Neural Networks
- Random Forests
- Regression
- Support Vector Machines
- K-Nearest Neighbors
- Naïve Bayes

- Example inputs, but no outputs; algorithm does all the work
- Clustering
- Self-Organizing Maps
- Association Analysis
- Singular Value Decomposition
  - JMP uses SVD as part of several routines

# Types of Machine Learning

## Supervised vs Unsupervised

- Example inputs and outputs provided; algorithm determines relationship(s)
- Decision Trees
- Neural Networks
- Random Forests
- Regression
- Support Vector Machines
- K-Nearest Neighbors
- Naïve Bayes

- Example inputs, but no outputs; algorithm does all the work
- Clustering
- Self-Organizing Maps
- Association Analysis
- Singular Value Decomposition
  - JMP uses SVD as part of several routines – prominently in Text Explorer

# The (Possible) Trade-Off

## Accuracy vs Interpretability

- I can walk through the path of the Decision Tree to make a prediction, but there's no "meaning" behind the cut-offs.
- The coefficients in an SVM model have no meaning; they are just used to obtain the predicted outcome.
- Can produce models with very high predictive accuracy.

- Regression models provide model coefficients that have inherently interpretable meaning.
- "This model says that if I increase this input by 1 unit, my response will go up 10 units!"
- Lasso and Elastic Net in GenReg bring a Machine Learning mindset to an explainable regression model.
- As discussed by Galit Schmueli, this is the Explain vs Predict problem.

# Does lack of interpretability matter?

## Maybe…it depends on the use case

- If the algorithm accurately predicts future events or gives desired outcomes, maybe that's all that matters.
- If stakeholders want to have more concrete answers as to how to "improve their score", an explainable model may be preferred.
- The key for any model is whether can you put the output into action.

# Computers win; goodbye human analysts

## Not so fast…

- There are lots of algorithms; which one(s) are best for this problem?
- What data should be included or excluded?
- Where is the point of diminishing returns?
- Are there inherent biases in the model?
- When should the model be updated?

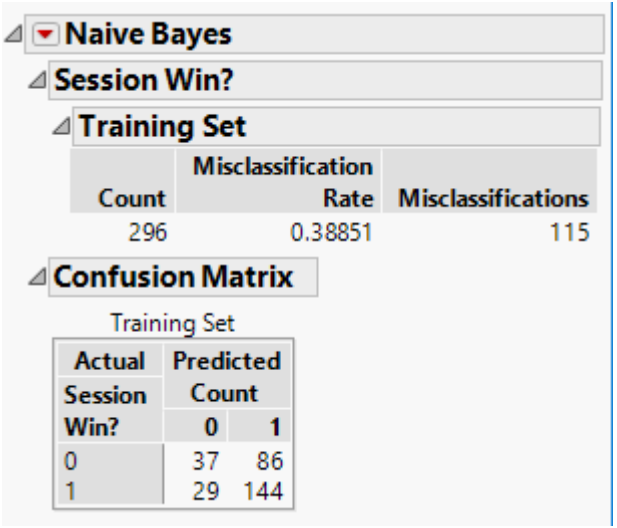
# Demo Time

## Supervised Learning

- Two older machine learning platforms have been updated recently – K-Nearest Neighbors and Naïve Bayes
- A brand new platform – Support Vector Machines (SVM)
  - Not to be confused with Structural Equation Modeling (SEM)

# Naïve Bayes

- Strong assumption that predictors are independent
- Calculates probability of class membership based on conditional probabilities given the level of the predictors
- Efficient algorithm; inefficient results (in my experience)



**Naive Bayes**

**Session Win?**

**Training Set**

| Count | Misclassification Rate | Misclassifications |
|---|---|---|
| 296 | 0.38851 | 115 |

**Confusion Matrix**

Training Set

| Actual Session Win? | Predicted Count 0 | 1 |
|---|---|---|
| 0 | 37 | 86 |
| 1 | 29 | 144 |

# K-Nearest Neighbors

- Predicts responses based on observations "nearby"
- Distances measured by Euclidean distance
- Continuous response – average of the k-nearest
- Categorical response – most frequent of the k-nearest

**K Nearest Neighbors**

**Session Win?**

**Training Set**

| K | Count | Misclassification Rate | Misclassifications | |
|---|---|---|---|---|
| 1 | 296 | 0.47635 | 141 | |
| 2 | 296 | 0.51689 | 153 | |
| 3 | 296 | 0.46959 | 139 | |
| 4 | 296 | 0.49324 | 146 | |
| 5 | 296 | 0.49324 | 146 | |
| 6 | 296 | 0.48649 | 144 | |
| 7 | 296 | 0.44257 | 131 | |
| 8 | 296 | 0.46284 | 137 | |
| 9 | 296 | 0.43919 | 130 | * |
| 10 | 296 | 0.45270 | 134 | |

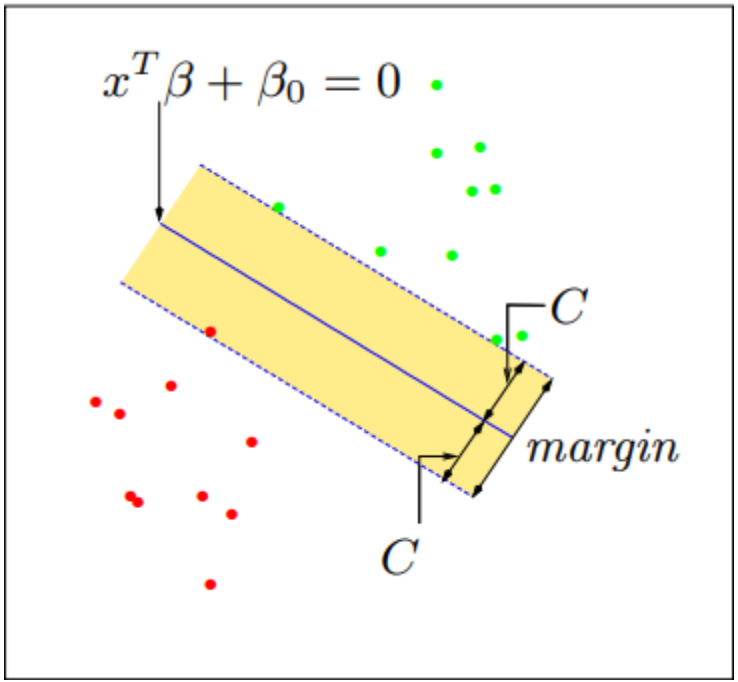
**Confusion Matrix for Best K=9**

Training Set

| Actual Session Win? | Predicted Count 0 | 1 |
|---|---|---|
| 0 | 45 | 78 |
| 1 | 52 | 121 |

# Support Vector Machines

- Maximum Margin Classifier
- Maximizes the space around the classification line to separate the classes
- Distance metric in the “Kernel”
  - Linear Kernel – Euclidean distance
  - RBF Kernel – Gaussian similarity measure
- Parameters of the kernel need to be chosen



# References


- Wikipedia. 2020. *Machine Learning.* https://en.wikipedia.org/wiki/Machine_learning. [Online: Accessed 18-May-2020].
- Encyclopaedia Britannica. 2020. *Machine Learning*. https://www.britannica.com/technology/machine-learning. [Online: Accessed 18-May-2020].
- Shmueli, G. 2010. *To Explain or To Predict?*, Statistical Science **25**, no. 3, 289–310.

# Thank you!

jmp.com