Boulder 2016 B-cycle

Data Exploration
Regression Machine Learning
Classification Machine Learning

Harpreet Bhasin

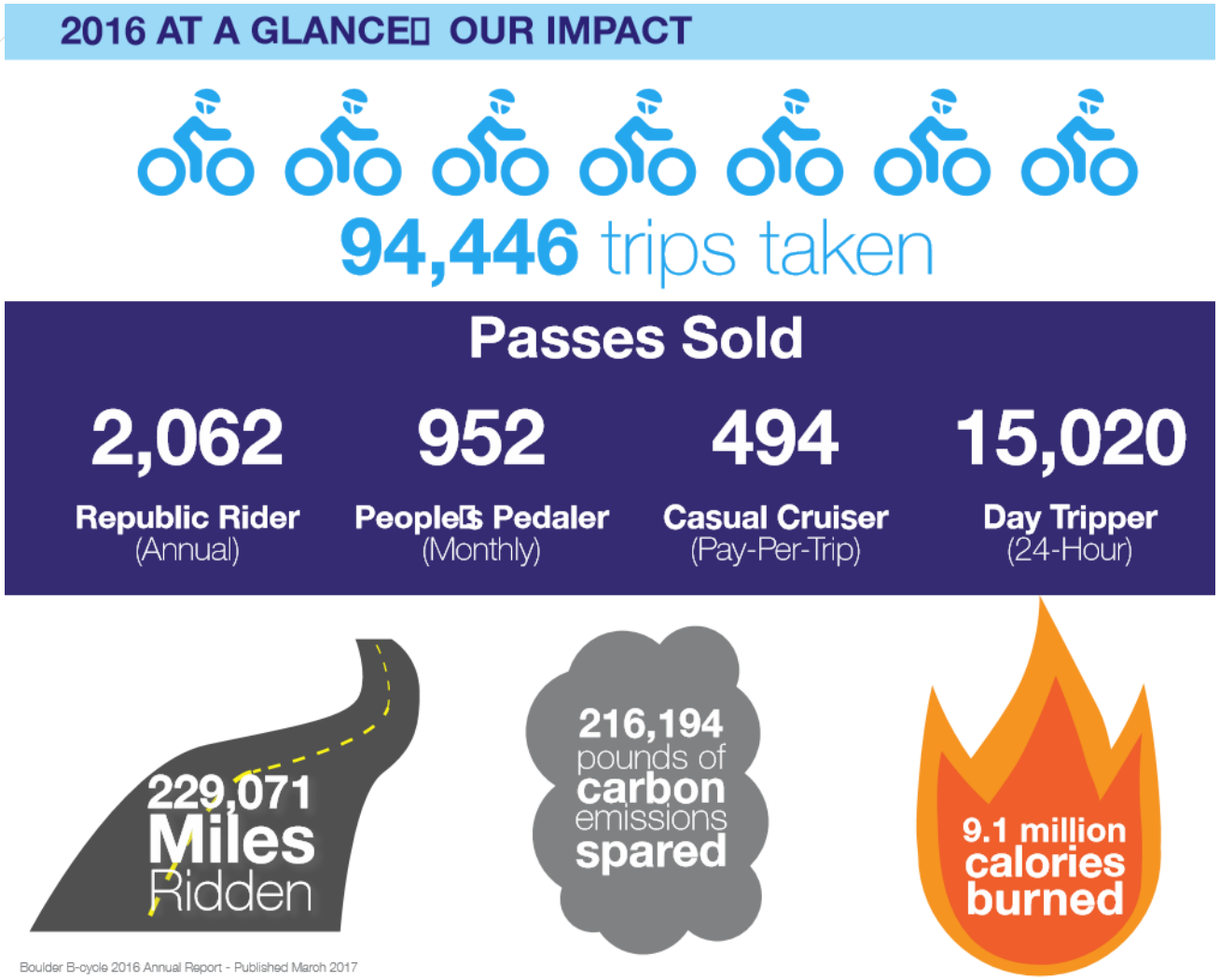
Boulder B-cycle

- Non-profit public organization
- Owns and operates an automated public bike sharing system
- Has 300 bicycles and 41 kiosks located throughout downtown Boulder and nearby areas
- Complements and integrates with Boulder's comprehensive metropolitan transportation
- Contributes to Boulder becoming the healthiest and greenest city in America
- Encourages the replacement of short car trips for recreational, social and functional purposes

Source: https://www.Boulderbcycle.com/company

Source: https://cdn01.bcycle.com/libraries/docs/librariesprovider35/default-document-library/2016-annual-report-final-3_13_web.pdf?sfvrsn=7ac85cc5_2

Boulder Bike Share

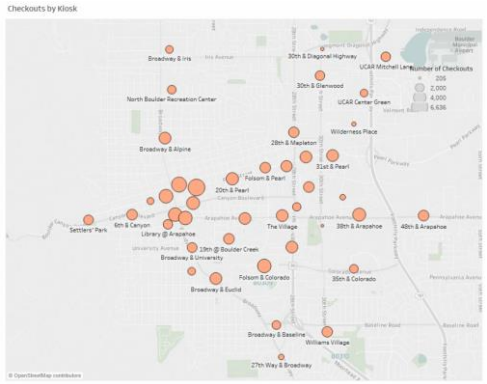


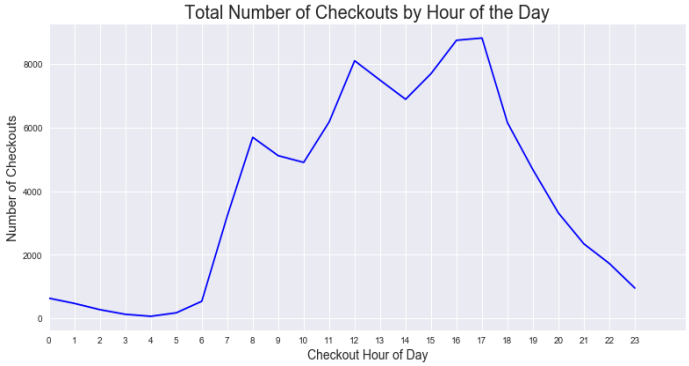
Source: https://Boulder.bcycle.com/docs/librariesprovider34/default-document-library/dbs_annualreport_2016_05.pdf

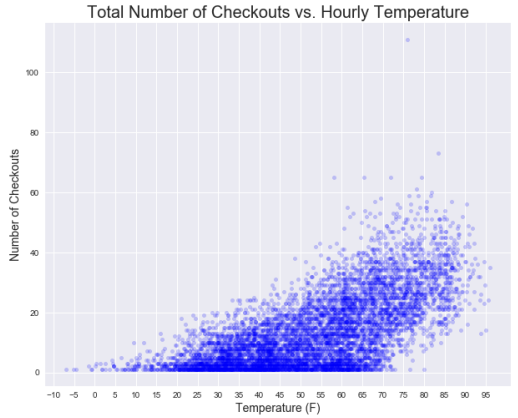
The Objective

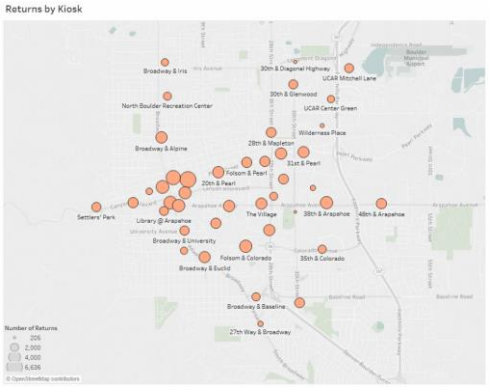
- Explore the publicly available 2016 Trips dataset and visualize the data to provide useful and interesting information
- Deploy a variety of regression machine learning models to predict number of bike checkouts using a combination of calendar, clock and weather attributes
- Deploy variety of classification machine learning models to predict number of bike checkouts using a combination of calendar, clock and weather attributes
- Provide and/or present findings to Boulder B-cycle executives to improve future ridership

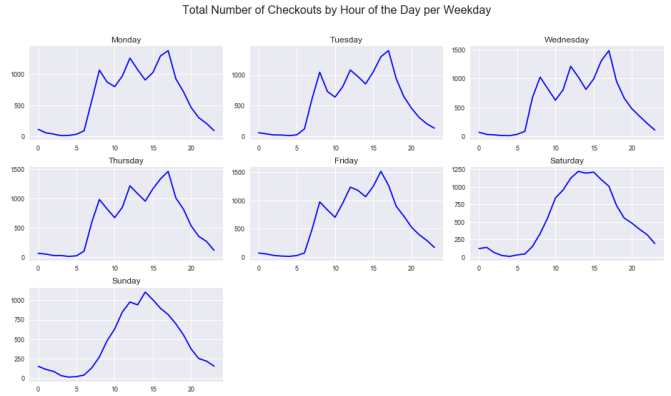
Step 1- Data Exploration

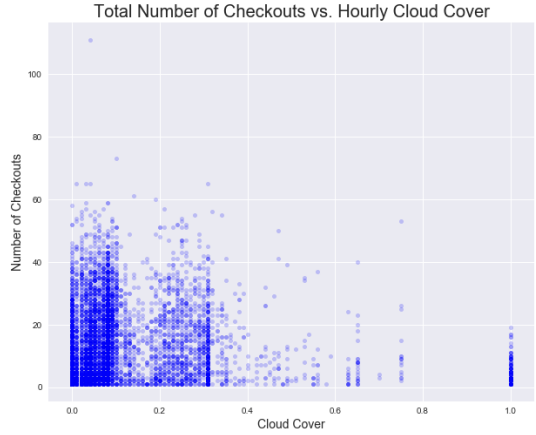












Step 2 – Regression Models

- Predict number of bike checkouts using the following models
  - Linear Regression
    - Most widely used statistical and machine learning technique to model relationship between two sets of variables typically using a straight line. Simple to use and fast performance but lacks high accuracy when compared to non-linear models.
  - Lasso Regression
    - A type of linear regression that uses shrinkage to reduce data values toward the mean. Well suited for automating feature selection.
  - Ridge Regression
    - Well suited for data that suffers from multicollinearity, i.e. features with high correlation.
  - Bayesian Ridge Regression
    - An approach to linear regression in which the statistical analysis is undertaken using Bayesian inference.
  - Decision Tree Regression
    - Uses a tree like structure to derive a final decision on the outcome of the analysis.
  - Random Forest Regression
    - An ensemble learning method that operates by constructing a multitude of decision trees to arrive at the mean prediction.
  - Extra Trees Regression
    - An extremely randomized tree regressor. Builds a totally random decision tree.
  - Nearest Neighbors Regression
    - A simple algorithm that uses a similarity measure (e.g. distance between neighbors) to predict the outcome.

Step 3 – Classification Models

- Predict number of bike checkouts using the following models
  - Logistic Regression Classification
    - Similar to linear regression but used for classification.
  - Decision Tree Classification
    - Uses a tree like structure to derive a final decision on the outcome of the analysis.
  - Random Forest Classification
    - Similar to random forest regression but used for classification.
  - Extra Trees Classification
    - Similar to extra trees regression but used for classification.
  - Naïve Bayes Classification
    - Uses the Bayes' Theorem (i.e. assumes that the presence of a particular feature is unrelated to the presence of any other feature).
  - Gradient Boosting Classification
    - A machine learning method that produces a prediction model in the form of an ensemble of weak prediction models, typically decision trees.
  - Nearest Neighbors Regression
    - Similar to nearest neighbors regressor but used for classification
  - Multi-Layer Perceptron Classification
    - A feedforward artificial neural network mode that maps sets of input data onto a set of appropriate outputs.

Results – Regression Models

Regression Modeling Summary – Categorical Feature Set

	Linear	Lasso	Ridge	Bayesian Ridge	Decision Tree	Random Forest	Extra Trees	Nearest Neighbors
Training Test Score	0.680	0.677	0.677	0.680	1.000	0.943	1.000	0.597
Test Set Score	0.676	0.674	0.674	0.676	0.423	0.679	0.699	0.496
R Squared	0.822253	0.821127	0.821127	0.822232	0.650078	0.824289	0.835802	0.70400
RMSE	46.268837	46.533272	46.533272	46.273848	82.481079	45.789939	43.059707	72.051003

- **Extra Trees is best performing regressor with 44 features**
- **Random Forest is pretty close**

Regression Modeling Summary – Numerical Feature Set

	Linear	Lasso	Ridge	Bayesian Ridge	Decision Tree	Random Forest	Extra Trees	Nearest Neighbors
Training Test Score	0.452	0.444	0.444	0.451	1.000	0.947	1.000	0.867
Test Set Score	0.470	0.463	0.463	0.471	0.481	0.735	0.739	0.616
R Squared	0.655919	0.680605	0.680605	0.685989	0.693454	0.857242	0.859465	0.784742
RMSE	74.533468	75.555613	75.555613	74.519813	73.070383	37.319948	36.782801	54.076287

- **Extra Trees is best performing regressor with 9 features**
- **Either Extra Trees or Random Forest regressor is recommended for 44 or 9 features**

Regressor Test on Unseen Samples

Sample Number	Actual Number of Checkouts	Predicted Number of Checkouts	+/-
1	12	12	0
2	48	48	0
3	9	9	0
4	33	33	0
5	12	12	0
6	13	13	0
7	9	9	0
8	6	6	0
9	8	8	0
10	5	5	0

- **Extra Trees Regressor with 85.9% accuracy predicted all 10 unseen samples accurately.**

Results – Classification Models - 1

Classification Modeling Summary – Categorical Feature Set

	Logistic	Decision Tree	Random Forest	Extra Trees	Naïve Bayes	Nearest Neighbors	Gradient Boosting	Multi-Layer Perceptron
Accuracy	0.877010	0.811258	0.856197	0.877010	0.756386	0.714286	0.817408	0.875591
F1 (macro)	0.877009	0.811243	0.856088	0.876795	0.747293	0.703119	0.817388	0.875557
F1 (micro)	0.877010	0.811258	0.856197	0.877010	0.756386	0.714286	0.817408	0.875591
Precision (macro)	0.877030	0.811341	0.857210	0.877409	0.799952	0.751804	0.817524	0.875965
Precision (micro)	0.877010	0.811258	0.856197	0.877010	0.756386	0.714286	0.817408	0.875591
Recall (macro)	0.8777014	0.811251	0.856172	0.876995	0.756566	0.714103	0.817399	0.875576
Recall (micro)	0.877010	0.811258	0.856197	0.877010	0.756386	0.714286	0.817408	0.875591
Cross Validation	0.858643	0.804258	0.856145	0.864045	0.740562	0.721885	0.799319	0.870338
Execution Time (sec)	10.791598	0.317235	4.023082	3.425960	0.175422	0.994537	48.096231	9.780759

- **Extra Trees Classifier is best performing classifier with 44 features.**
- **Random Forest Classifier is pretty close.**

Results – Classification Models - 2

Classification Modeling Summary – Numerical Feature Set

	Logistic	Decision Tree	Random Forest	Extra Trees	Naïve Bayes	Nearest Neighbors	Gradient Boosting	Multi-Layer Perceptron
Accuracy	0.772942	0.845790	0.878430	0.872280	0.771523	0.815043	0.851939	0.867550
F1 (macro)	0.772932	0.845788	0.878403	0.872190	0.769910	0.812961	0.851927	0.867515
F1 (micro)	0.772942	0.845790	0.878430	0.872280	0.771523	0.815043	0.851939	0.867550
Precision (macro)	0.773004	0.845802	0.878721	0.873266	0.779530	0.829477	0.852033	0.867893
Precision (micro)	0.772942	0.845790	0.878430	0.872280	0.771523	0.815043	0.851939	0.867550
Recall (macro)	0.772949	0.845790	0.878416	0.872256	0.771603	0.819944	0.851932	0.867535
Recall (micro)	0.772942	0.845790	0.878430	0.872280	0.771523	0.815043	0.851939	0.867550
Cross Validation	0.771672	0.832108	0.874368	0.871019	0.765200	0.822538	0.843486	0.852682
Execution Time (sec)	9.676564	0.301543	3.972808	3.246149	0.106590	0.953759	20.225848	4.669855

- **Random Forest is best performing classifier with 9 features.**
- **Extra Trees Classifier is pretty close.**

Classifier Test on Unseen Samples

Sample Number	Actual Number of Checkouts	Class Number	Predicted Number of Checkouts	Class Number
1	Greater than 10	1	Greater than 10	1
2	Greater than 10	1	Greater than 10	1
3	Between 1 and 10	0	Between 1 and 50	0
4	Greater than 10	1	Greater than 10	1
5	Greater than 10	1	Greater than 10	1
6	Greater than 10	1	Greater than 10	1
7	Between 1 and 10	0	Greater than 10	1
8	Greater than 10	1	Greater than 10	1
9	Between 1 and 10	0	Between 1 and 10	1
10	Between 1 and 10	0	Between 1 and 10	0

- **Random Forest Classifier with 87.8% accuracy predicted 9 out of 10 unseen samples accurately.**
- **Of the remaining 1 sample, it predicted one class above the actual class.**

Use of Regressor and Classifier for Boulder B-cycle

- Help predict number of bike checkouts in 2017 based on 2016 trips dataset.
- Drill down on calendar variables (month and weekday), clock variable (checkout hour) and weather variables (temperature, cloud clover, humidity, wind speed and visibility) to predict number of bike checkouts in 2017.
- Optimize number of available bikes at checkout kiosks.
- Use the Extra Trees Regressor to predict a number of bike checkouts.
- Use the Random Forest Classifier to predict number of bike checkouts within a range (1 to 10 and greater than 10).

Next Steps

- Develop a simple desktop, web or mobile app that takes calendar, clock and weather variables as inputs and predicts the number of checkouts as the output.
- Longmont, CO has just introduced its bike sharing system – this study could be useful to the management.