Denver 2016 B-cycle

Data Exploration
Regression Machine Learning
Classification Machine Learning

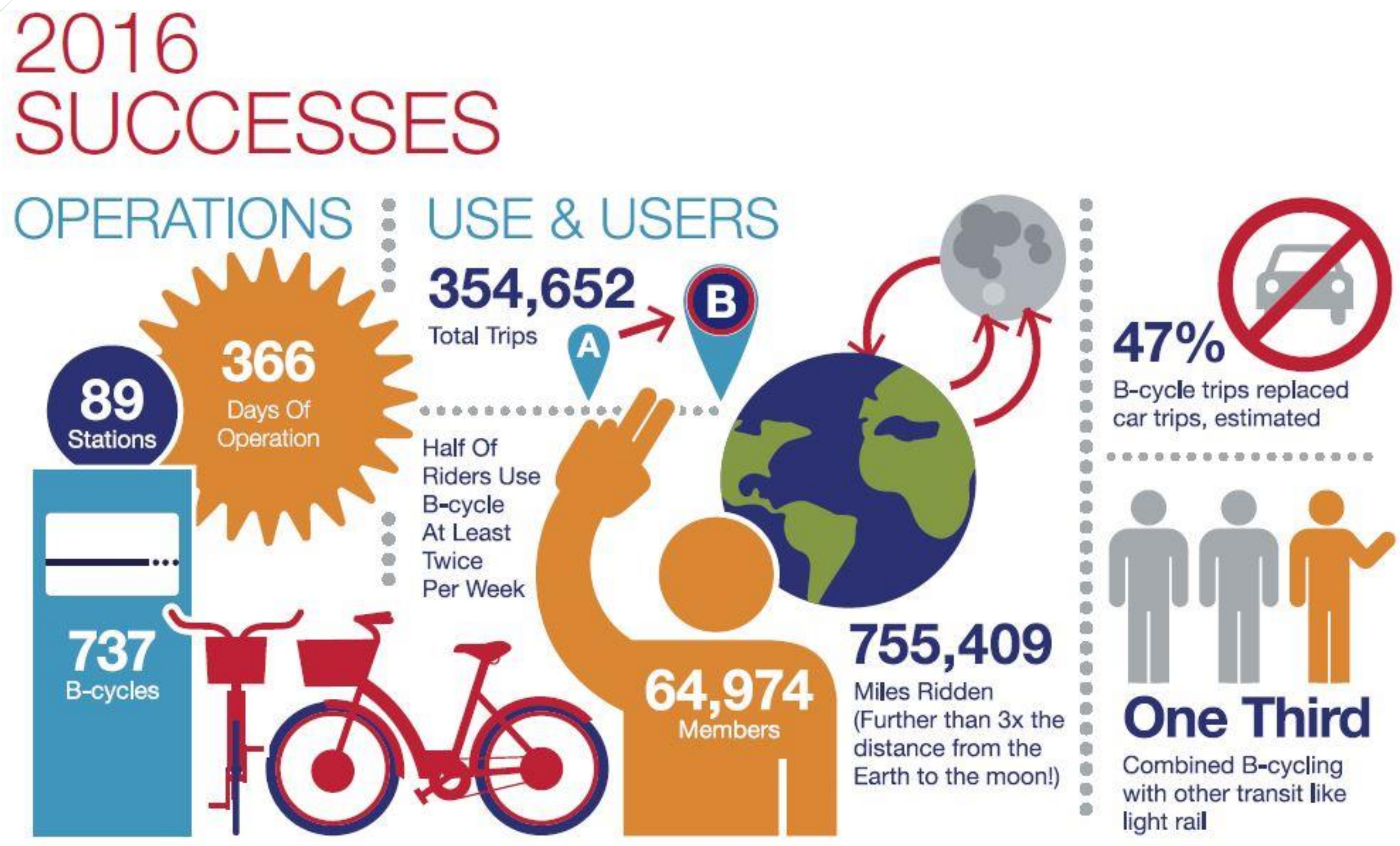
Harpreet Bhasin

Denver B-cycle

- Non-profit public organization
- Owns and operates an automated public bike sharing system
- Has 737 bicycles and 89 kiosks located throughout downtown Denver and nearby areas
- Complements and integrates with Denver's comprehensive metropolitan transportation
- Contributes to Denver becoming the healthiest and greenest city in America
- Encourages the replacement of short car trips for recreational, social and functional purposes

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

Denver Bike Share

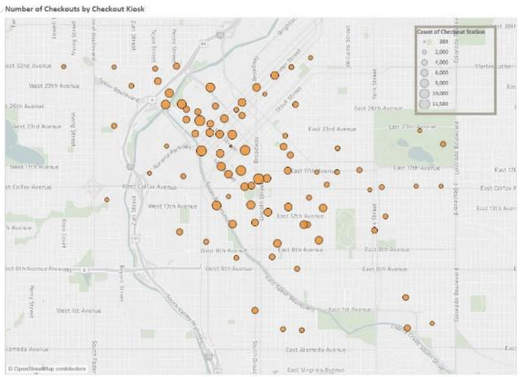


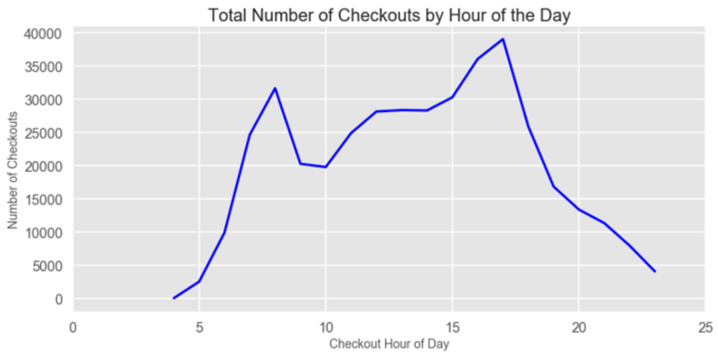
Source: https://denver.bcycle.com/docs/librariesprovider34/default-document-library/dbs_annualreport_2016_05.pdf

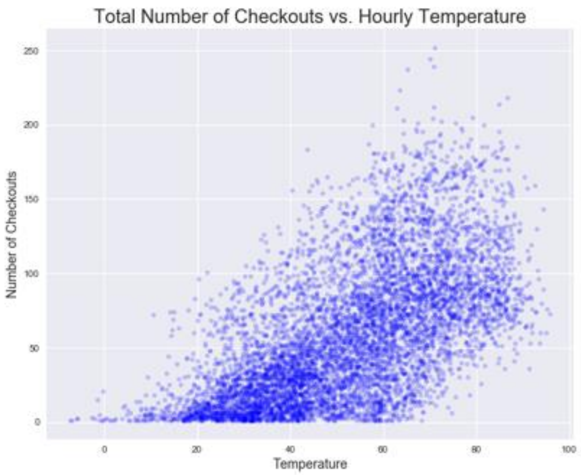
The Objective

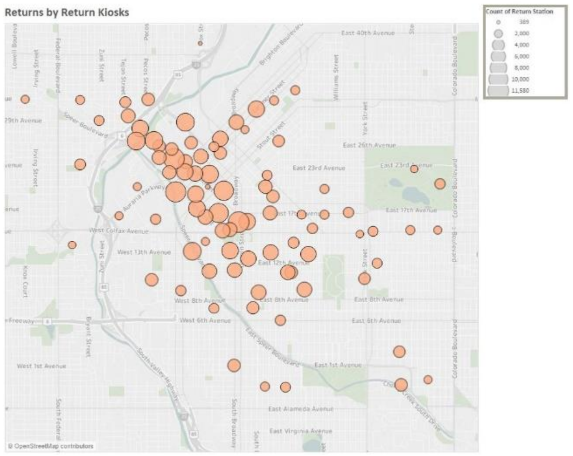
- Explore the publicly available 2016 Trips dataset and visualize the data to provide useful and interesting information
- Deploy a variety of regression machine learning models to predict number of bike checkouts using a combination of calendar, clock and weather attributes
- Deploy variety of classification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

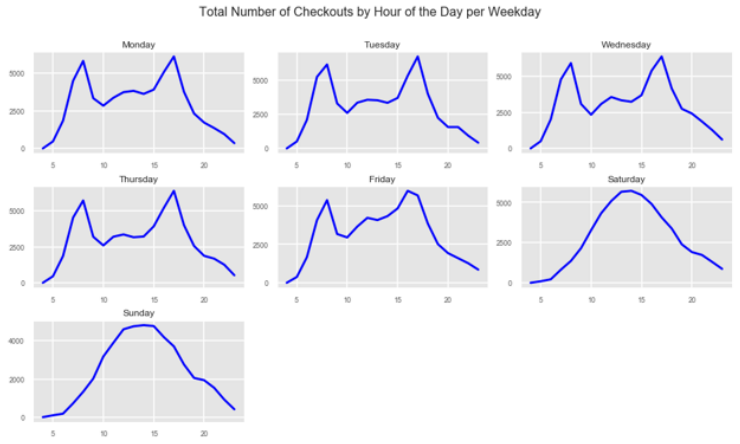
Step 1- Data Exploration

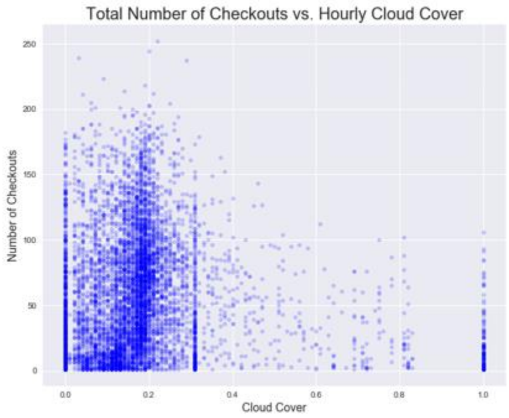












Step 2 – Regression Models

- 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	0.676	0.676	0.676	0.676	1.000	0.969	1.000	0.575
Test Set Score	0.696	0.696	0.696	0.696	0.718	0.825	0.840	0.476
R Squared	0.834519	0.834457	0.834457	0.834448	0.847276	0.908443	0.916278	0.690249
RMSE	627.95439	628.16826	628.16826	628.19832	583.57445	361.43485	331.86035	1082.98114

- **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	0.433	0.433	0.433	0.433	1.000	0.975	1.000	0.880
Test Set Score	0.448	0.447	0.447	0.447	0.741	0.854	0.838	0.646
R Squared	0.669090	0.668243	0.668243	0.668785	0.861079	0.924077	0.915609	0.803447
RMSE	1142.475	1144.818	1144.818	1143.319	534.800	302.172	334.397	733.229

- **Random Forest 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	92	96	+4
2	12	13	+1
3	55	56	+1
4	111	112	+1
5	76	72	-4
6	41	37	-4
7	8	14	+6
8	81	99	+18
9	65	64	-1
10	14	14	0

- **Random Forest Regressor with 92.4% accuracy predicted 9 out of 10 unseen samples well within 7.6%.**
- **Of the remaining 1 sample, its prediction was off by 22%.**

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	0.662281	0.640838	0.705653	0.712693	0.347466	0.545809	0.665205	0.689571
F1 (macro)	0.500343	0.535669	0.580615	0.597661	0.277657	0.316936	0.540916	0.488650
F1 (micro)	0.662281	0.640838	0.705653	0.712693	0.347466	0.545809	0.665205	0.689571
Precision (macro)	0.547424	0.534911	0.611702	0.609287	0.353727	0.426109	0.551274	0.600202
Precision (micro)	0.662281	0.640838	0.705653	0.712693	0.347466	0.545809	0.665205	0.689571
Recall (macro)	0.488317	0.538071	0.565099	0.590623	0.399370	0.314355	0.532883	0.514250
Recall (micro)	0.662281	0.640838	0.705653	0.712693	0.347466	0.545809	0.665205	0.689571
Cross Validation	0.654620	0.639123	0.695088	0.708480	0.350292	0.552339	0.652398	0.665322
Execution Time (sec)	15.989999	0.337216	4.064112	3.654782	0.211667	0.980936	137.870316	9.564595

- **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	0.571637	0.654483	0.702242	0.700780	0.462476	0.608187	0.670565	0.585283
F1 (macro)	0.317235	0.536431	0.565431	0.571989	0.288432	0.400182	0.551396	0.314778
F1 (micro)	0.571637	0.654483	0.702242	0.700780	0.462476	0.608187	0.670565	0.585283
Precision (macro)	0.329322	0.534468	0.605005	0.591664	0.366608	0.462348	0.567401	0.368854
Precision (micro)	0.571637	0.654483	0.702242	0.700780	0.462476	0.608187	0.670565	0.585283
Recall (macro)	0.330366	0.538836	0.549900	0.560451	0.331956	0.392413	0.539238	0.342865
Recall (micro)	0.571637	0.654483	0.702242	0.700780	0.462476	0.608187	0.670565	0.585283
Cross Validation	0.573450	0.649181	0.691696	0.690877	0.470760	0.605263	0.660643	0.580409
Execution Time (sec)	14.257541	0.217554	4.202843	3.353612	0.128580	0.957223	76.372264	4.313864

- **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	Between 51 and 100	1	Between 51 and 100	1
2	Between 1 and 50	0	Between 1 and 50	0
3	Between 51 and 100	1	Between 1 and 50	0
4	Between 101 and 150	2	Between 101 and 150	2
5	Between 51 and 100	1	Between 51 and 100	1
6	Between 1 and 50	0	Between 1 and 50	0
7	Between 1 and 50	0	Between 1 and 50	0
8	Between 51 and 100	1	Between 1 and 50	0
9	Between 51 and 100	1	Between 51 and 100	1
10	Between 1 and 50	0	Between 1 and 50	0

- **Random Forest Classifier with 79.3% accuracy predicted 8 out of 10 unseen samples accurately**
- **Of the remaining 2 samples, it predicted one class below the actual class in both samples.**

Use of Regressor and Classifier for Denver 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 Random Forest Regressor to predict a number of bike checkouts.
- Use the Random Forest Classifier to predict number of bike checkouts within a range (1 to 50, 51 to 100, 101 to 150 and 151 to 252).

Next Steps

- Undertake similar project for Boulder B-cycle
- Longmont, CO has just introduced its bike sharing system – this study could be useful