Mountain Bike (MTB) Categorization Analysis

Michael Czerwinski & Justin Schulberg

2022-04-17

/ Introduction

// Project Overview

For this project, our team will determine whether the specifications of mountain bikes (MTB) are enough to differentiate between the different types of mountain bike categories.

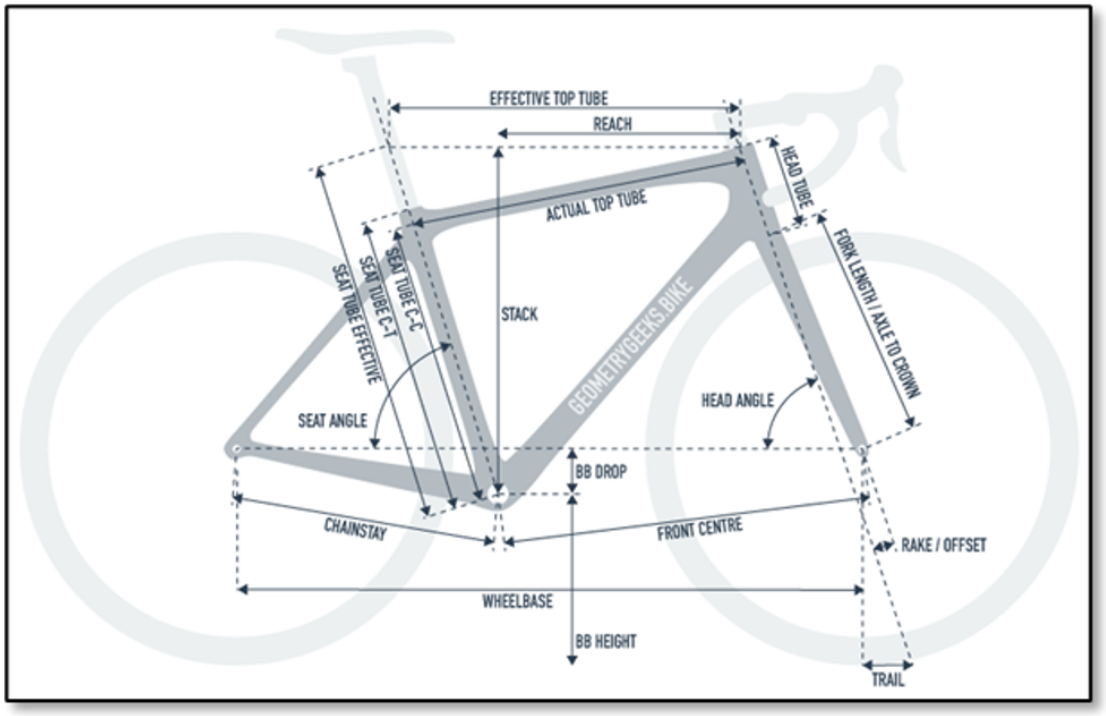
Currently, full suspension mountain bikes come in multiple categories:

- **Cross Country (XC)** | Tend to be the most lightweight, nimble, and designed to put the rider in an efficient pedaling position
- **Enduro (EN)** | Heavier frames, more travel and more downhill oriented geometry
- **Trail (TR)** | The most common category of bikes, considered to be the halfway point between XC and Enduro
- **All Mountain (AM)** | A more niche category which some manufacturers claim to be more downhill focused than trail bikes, but not designed for downhill races like Enduro bikes are
- **Downcountry (DC)** | A relatively new category between XC and Trail. Similar to the All Mountain category, these bikes aren't race specific like XC bikes tend to be, but are lighter and faster than trail bikes.

With all of the factors to consider when designing a bike, there are no clear boundaries between these categories. For example, one brand's Downcountry bike could be what another brand considers a Trail bike. The popular mountain biking website PinkBike has done in-depth analyses of many bikes across all categories, and covered the topics of which category bikes should be classified as and of how many categories is sufficient, as seen in the video here.

The goal of our project is to determine how many, if any, discrete categories should exist for mountain bikes. Since most specifications and geometric measurements have one direction when moving across the spectrum of bikes, it's reasonable to believe that these measurements could be reduced to much fewer dimensions, and perhaps even one continuous principle component rather than discrete categories. We can also cluster the bikes together based on some of these specifications and geometric measurements.

As an example, here is a diagram of some of the different types of geometric specifications on mountain bikes:



Various Dimension Features of a Bike's Geometry

// The Data

The data was retrieved manually from each of the mountain bike company's websites. All data and associated code for this project is available at the team's corresponding GitHub repository. Let's take a look at the data.

Model	Brand	Build Type	Price	Url	Image
element	rocky mountain			https://bikes.com/collections/element	
instinct	rocky mountain	Carbon 90	9799	https://bikes.com/collections/instinct	
Altitude	Rocky Mountain	Carbon 90 Rally Edition	10229	https://bikes.com/collections/altitude	
jeffsy	yt	Uncaged 6	9499	https://us.yt-industries.com/products/bikes/465/jeffsy-uncaged-6/preview	https://cdn-prod.yt-industr
izzo	yt	Uncaged 7	7499	https://us.yt-industries.com/products/bikes/466/izzo-uncaged-7/preview	https://cdn-prod.yt-industr

/ EDA

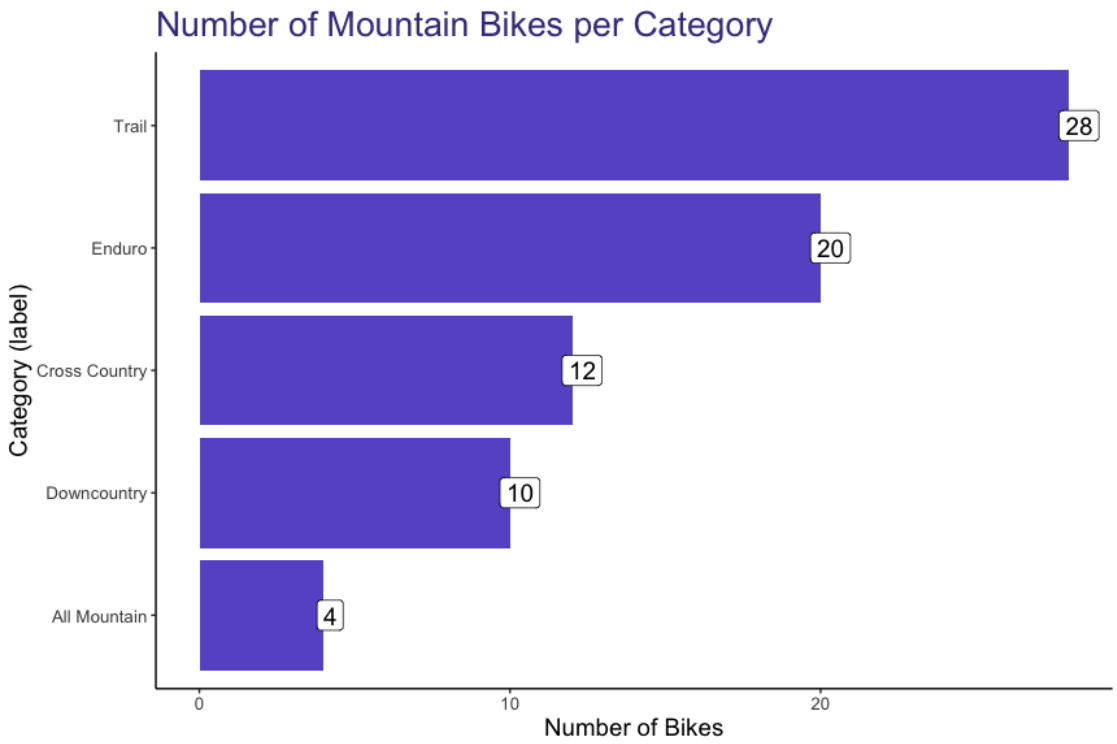
In this section, we'll take a look at the 74 mountain bikes in our dataset and some of the 27 features. We'll try to break down our understanding of the data in terms of `label`, our target variable that acts as the category for each mountain bike.

// Label (Mountainbike Category)

As stated earlier, there are 5 mountain bike categories in our dataset:

1. Cross Country (xc)
2. Enduro (en)
3. Trail (tr)
4. All Mountain (am)
5. Downcountry (dc)

Let's look at how many of each we have in our dataset.

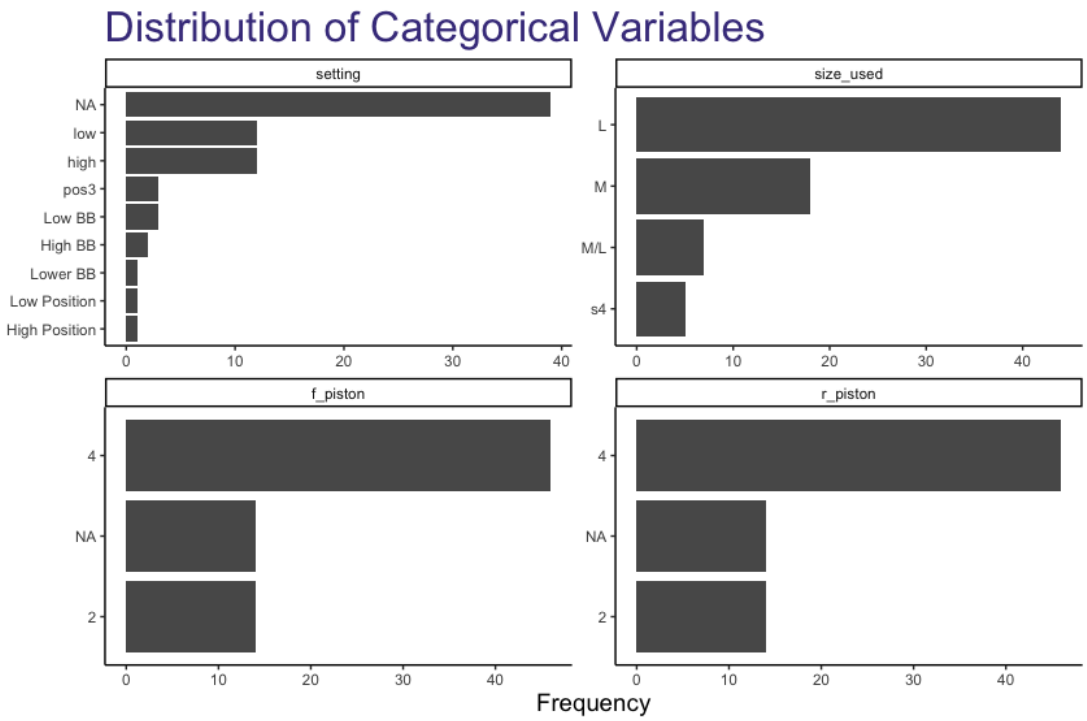


We see that out of our 74 bikes, most of them are Trail bikes, with the smallest grouping of bikes being all mountain bikes

// Categorical Variables

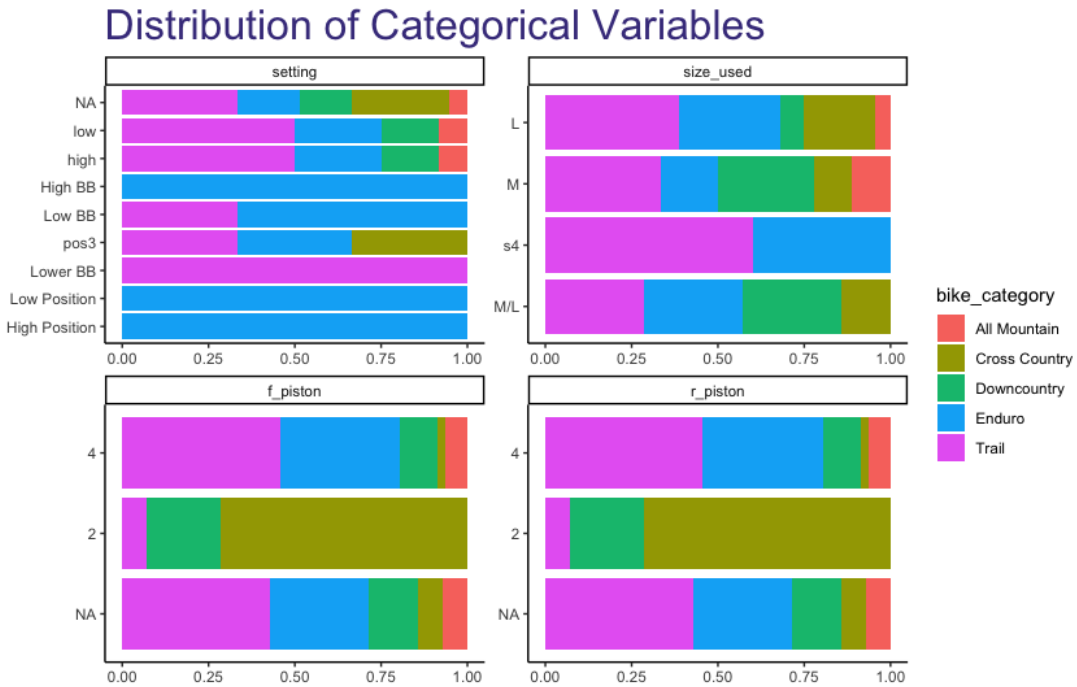
There are 4 categorical variables we'll take a look at to better understand our data:

1. Setting
2. Size
3. Front Piston (`f_piston`)
4. Rear Piston (`r_piston`)



- We see that only few bikes have a setting value, which is a feature that allows a rider to slightly adjust the frame's geometry to hone in rider comfort. Later on, we'll group by settings for the same bike and average the results to get a more accurate representation of the bikes' specs.
- Most of the bikes analyzed have 4 rear/front pistons. The two variables seem to be perfectly in-sync, leading us to believe that they're highly correlated.

But, really, we care about understanding how these different variables interact with our target variable, `label`. Let's look at their distribution and look for any patterns.



Here we see:

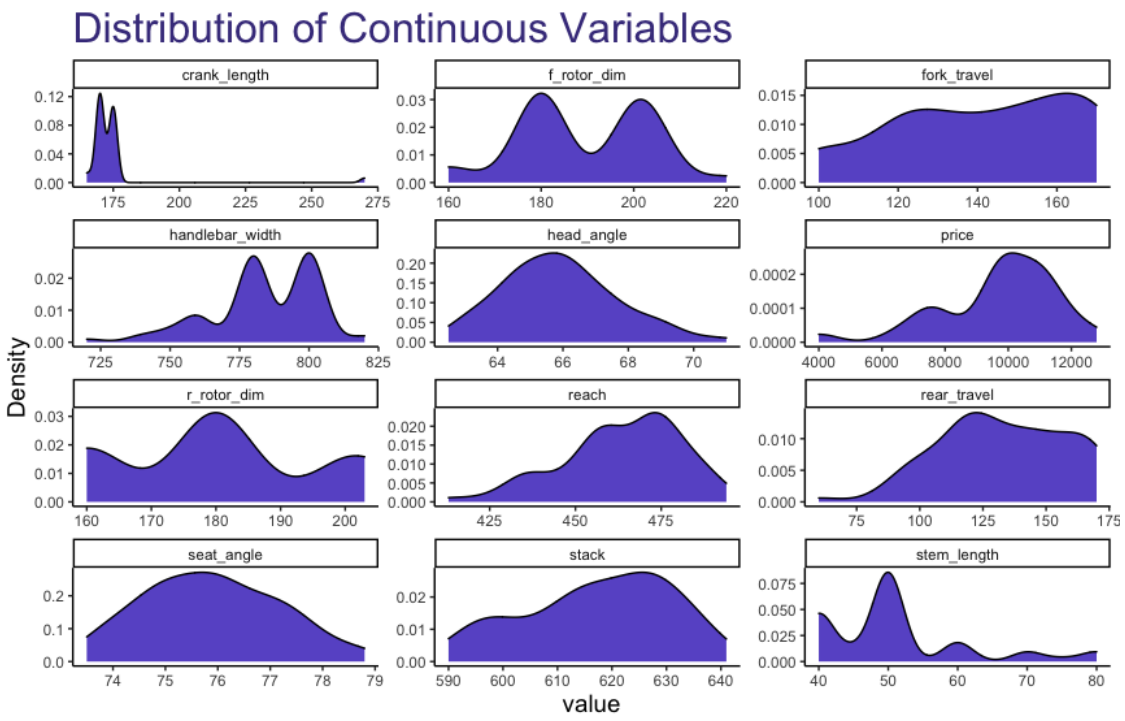
- The size used for most of the bikes is pretty evenly distributed. For the most part, we attempted to find bikes that are sized to the heights of the authors of this report (approx. 5'8"-5'11"), which

tended to be Large-sized bikes; however, for some bikes, like Trail, the specific bike's company website from which we pulled the data recommended a Medium-sized bike.

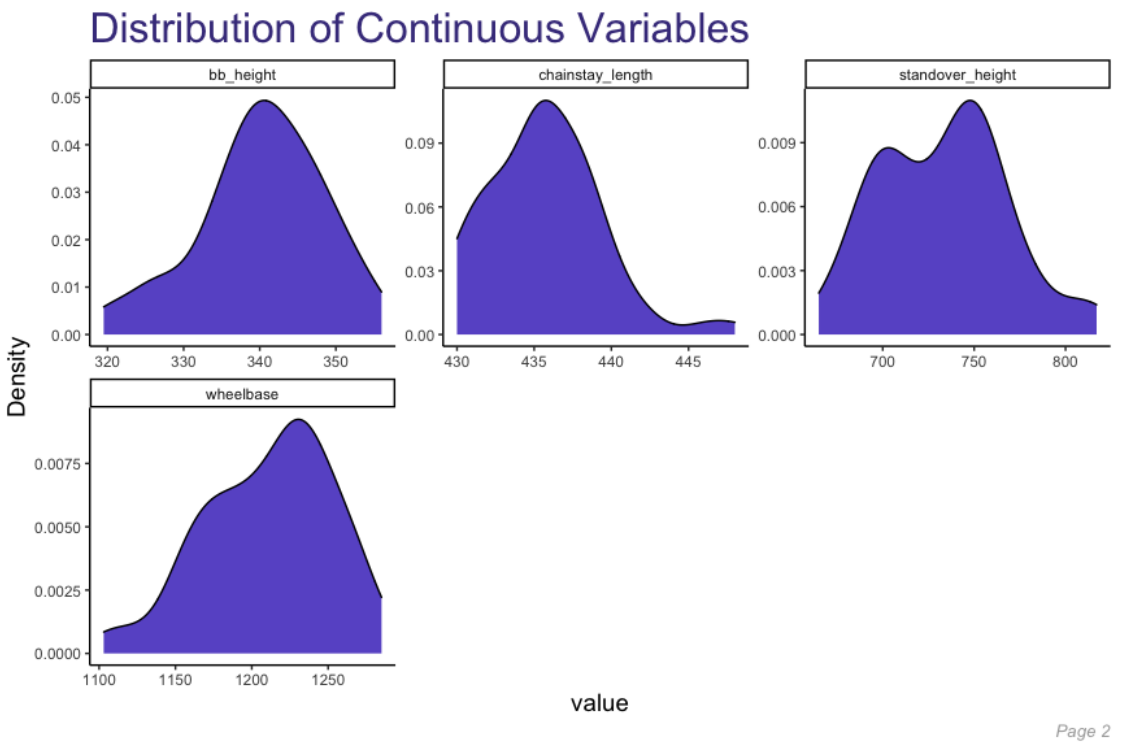
- Although most of the bikes have 4-piston brakes, of the bikes that have 2 pistons, most are Cross Country (*xc*) bikes. 4-piston brakes are known to have higher stopping power which is more important the more the rider intends to ride downhill. However, they come at the cost of additional weight, which most XC riders will avoid at all costs.

// Continuous Variables

To analyze the continuous features within our dataset, we built density plots for each of them to better understand their distribution.



Page 1



Page 2

/// ~Normally Distributed Variables:

- Chainstay_length
- Fork_travel
- Bb_height
- Seat_angle

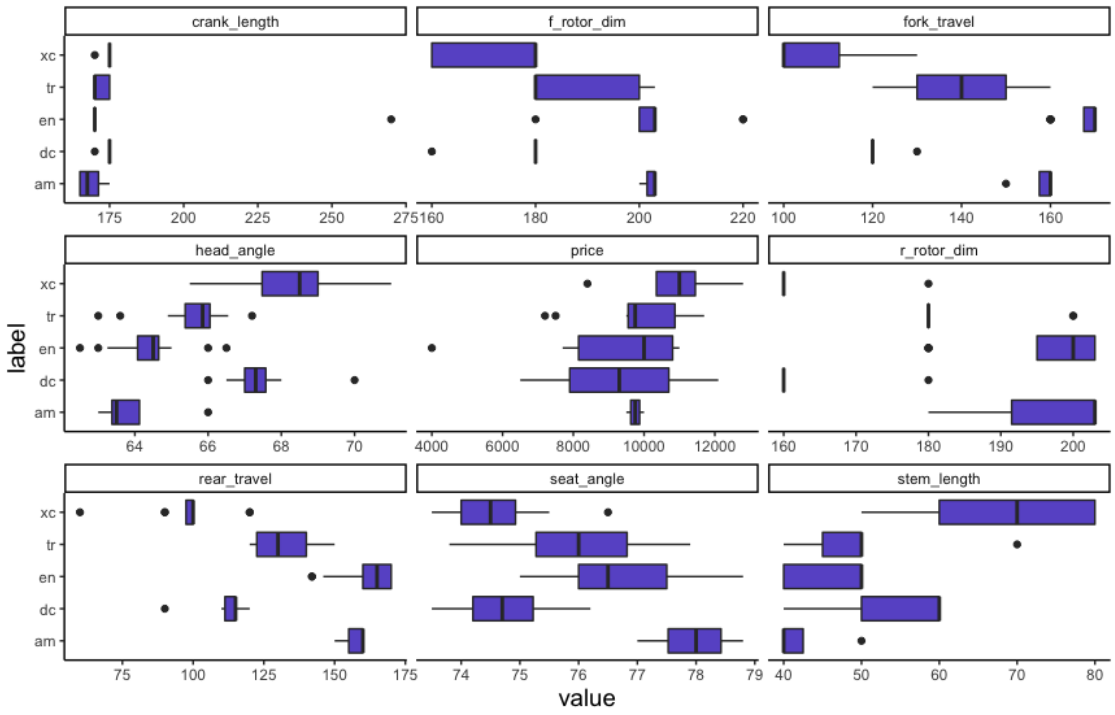
/// Skewed Variables:

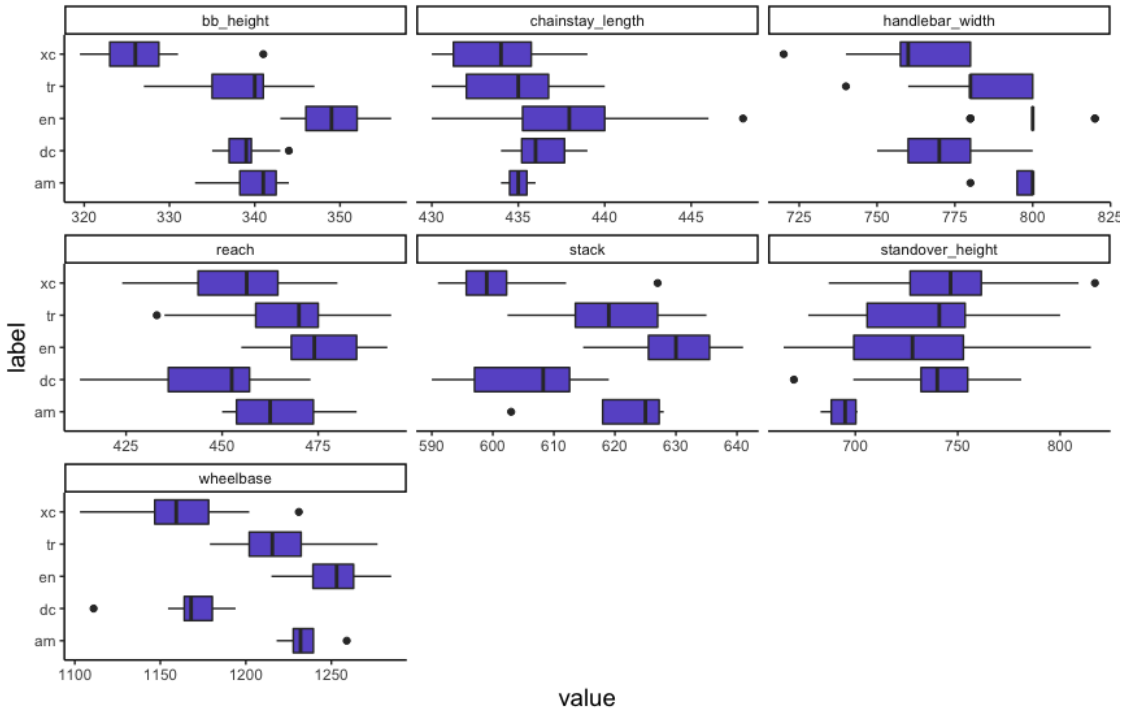
- Head_angle (skewed right)
- Handlebar_width (skewed left)
- Wheelbase (skewed left)

/// Multi-Modal Distributed Variables:

- f_rotor_dim / r_rotor_dim
- Stem_length

Like we did for continuous variables, let's look at the distribution of each of these predictors by our target variable, `label`, to look for any discernible patterns.





Here we see:

- Cross Country (*xc*) bikes tend to have the largest head angle and smallest seat angle compared to other bikes. They also have the largest stem length by a significant margin. Overall, Cross Country bikes tend to be the most differentiable from other bike categories;
- All Mountain (*am*) bikes have a significantly smaller standover height and, along with Enduro (*en*) bikes, have a much larger reach than other bike categories;
- As is generally expected, Trail (*tr*) bikes tend to fit mostly in the middle for most of these continuous' variables. This makes sense given that they tend to split the difference between Cross Country and Enduro bikes.

// Average bikes by flip-chip setting

Because some bikes' websites would have two different "settings" for the same-sized bike, we opted to include both options and average the two together to get one middle-of-the-road estimate for that type of bike. We end up performing this operation for 47% of the bikes in our dataset.

/ Methodology

Now that we have a better understanding of our mountain bike dataset, we'll formulate a plan to prove the following hypothesis:

> **Applying our own clustering algorithms will either give us a different set number of clusters (rather than the 5 pre-ordained categories) OR will not provide clearly defined clusters, leading us to believe that the bikes are actually created on a spectrum and cannot be grouped into one of the 5 pre-ordained categories.**

To do so, we'll:

- Try to use various methods to reduce the featureset and see if there are certain variables that can better be used to differentiate between different mountain bike categories.

- Apply various clustering and classification algorithms, including K-Means Clustering, Gaussian Mixture Models, and Multi-class Support Vector Machine, to disprove the notion that 5 distinct categories of Mountain Bikes exist.

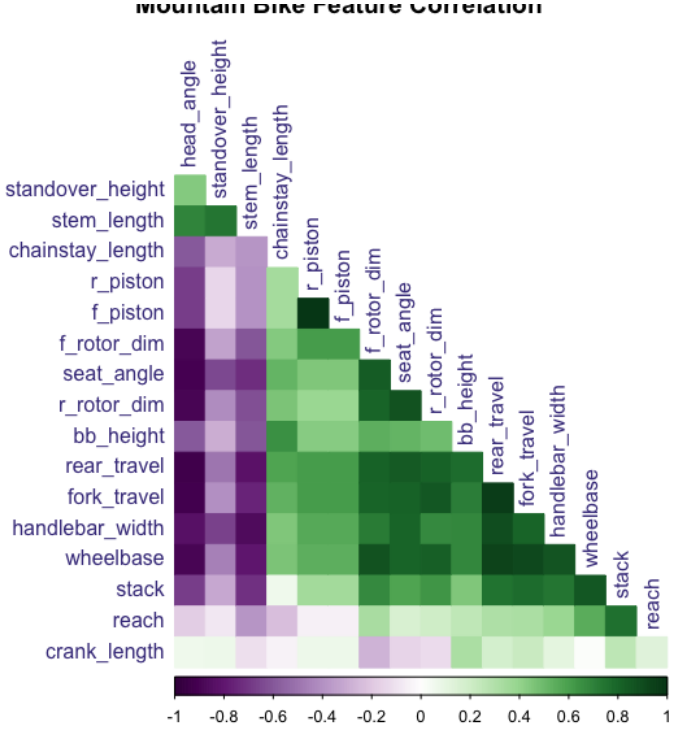
/ Variation Amongst Featureset

The first thing we'll do is look to see if any of the features in our dataset are better at explaining the variation amongst the different bikes than other features. That is, it's completely possible that two features are similar and don't have much variation in them, even across some of the different bike categories. To do so, we'll:

1. Look for highly correlated features and flag these for potential removal;
2. Run Principal Component Analysis (PCA) to see if certain features are better at explaining the variation in our data better than others.

// 1. Correlation

First, let's take a look at our most highly correlated features. We'll use the `corrplot()` function to better order the highly correlated features by the angular order of their eigenvectors.



Here we see some obvious correlations, for example:

- `f_piston` (front brakes) is perfectly correlated with `r_piston` (rear brakes), which makes sense since mountain bikes tend to use the same types/spec of brakes for the front vs. rear tires.
- `fork_travel` has a correlation above .95 with: c("rear_travel", "fork_travel"). This make sense; for example, `rear_travel` *should* be highly correlated with `fork_travel`.

In all, here are the most highly correlated variables (i.e. variables which have a correlation above .9 or below -.9):

variable	correlated_variable	correlation
f_piston	r_piston	1

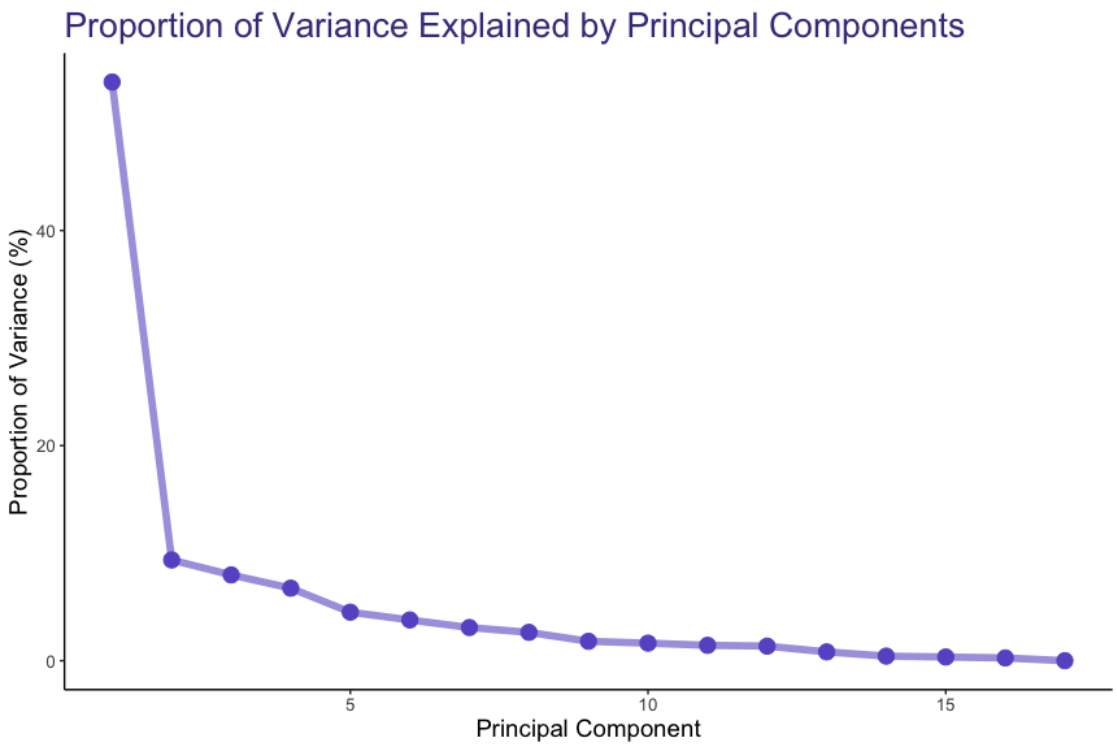
variable	correlated_variable	correlation
rear_travel	fork_travel	0.9608
rear_travel	wheelbase	0.9301
rear_travel	head_angle	-0.9219
fork_travel	wheelbase	0.9195
fork_travel	head_angle	-0.9193
head_angle	seat_angle	-0.9031

There are a solid amount, especially given that we only have 18 continuous columns in our dataset! For now, we'll opt to include everything. But later on, as we analyze the importance of different features, we'll look to remove some of the above variables first.

// 2. Principal Component Analysis (PCA)

Next, we'll apply PCA to our dataset. In so doing, we'll have to center and scale our data given how different the ranges are for certain measurements. Let's take a look at our 5 principal components which explain the largest proportion of variance in the data:

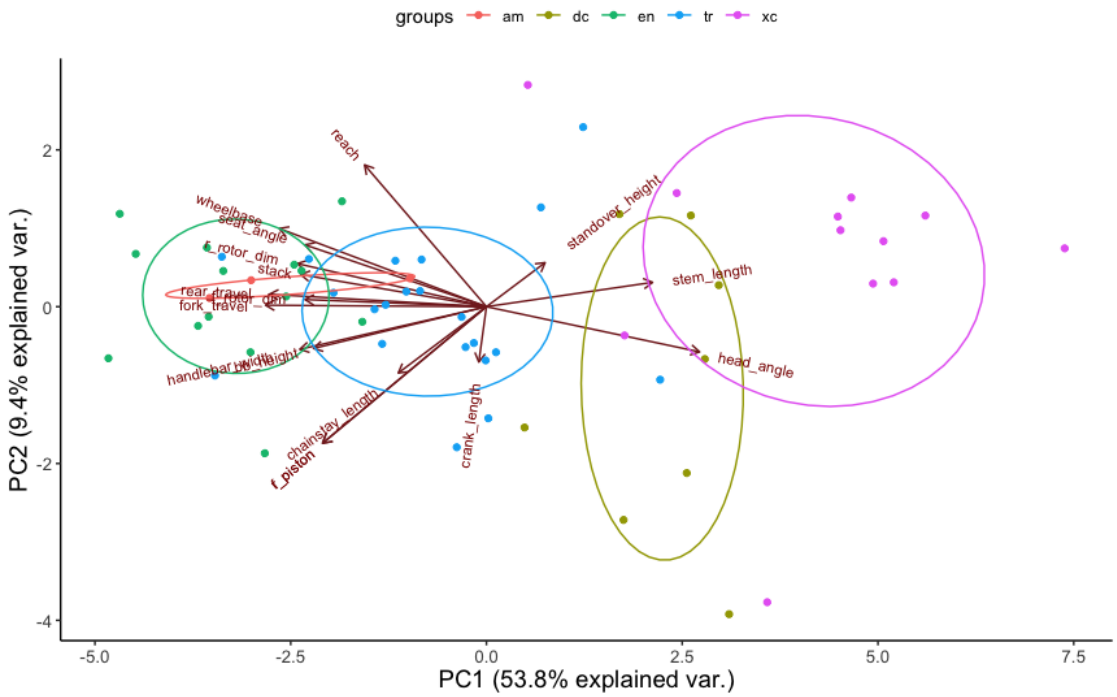
	PC1	PC2	PC3	PC4	PC5
Standard deviation	3.024	1.262	1.164	1.071	0.8761
Proportion of Variance	0.538	0.09369	0.07977	0.06745	0.04515
Cumulative Proportion	0.538	0.6317	0.7115	0.7789	0.8241



We can see that, actually, our 1st principal component alone explains more than half our data. Starting at the 2nd principal component, there's a distinguishable elbow point. After that, we have a huge drop-off. Starting at our 5th principal component, nearly 82.4% of the data's variation is properly explained.

This leads us to believe that the majority of the variation in our data can be explained by using just 1 principal component!

Let's take a look at how our top 2 principal components explain the 5 different mountain bike categories:



Here we can see that our top 2 principal components, which explain roughly 63.2% of the variation in our data, are already pretty good representations for describing the different components in our dataset. Even so, the groupings are distinctly plotted on the 2-D graph and it is pretty easy to see how the different bike categories (denoted by color) can be explained using a linear transformation of our existing data.

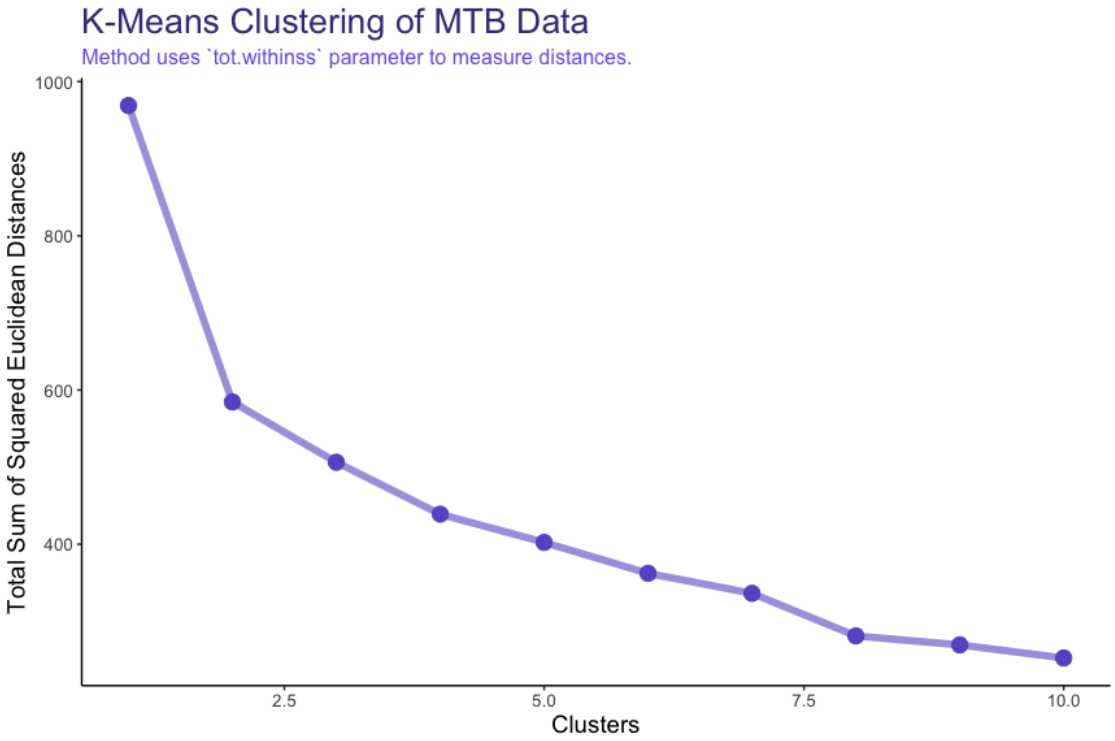
/ Clustering

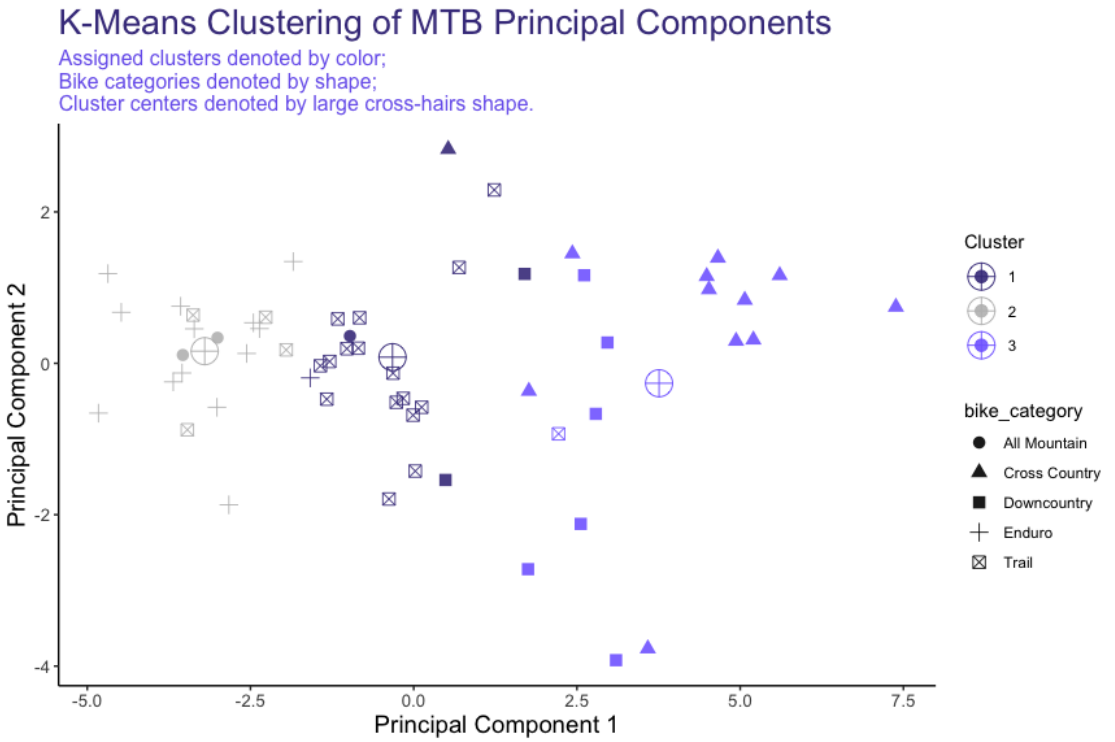
Because we are investigating the validity of mountain bike categories, one approach is to treat this dataset as unsupervised, stripping the bikes of their `label` and seeing if various clustering algorithms can re-create the 5 distinct `label`s. To do so, we'll take a look at the following algorithms:

- K-Means
- Gaussian Mixture Models (GMM)
- Support Vector Machine (SVM)

// K-Means

We'll start by using the K-Means Clustering algorithm, looking at various numbers of clusters (*k*) and seeing if the bikes logically group together.





Above, we attempted to graph the 3 clusters created using top 2 principal components in our data. For example, we can see Cluster #1 on the right-hand side of the chart, mostly composed of Cross Country bikes (triangles in the chart) and some Downcountry bikes (denoted by squares). Downcountry bikes also seem to be part of Cluster #2 (gray points), along with Trail bikes (denoted by squares with an 'x' in them) and some Enduro bikes (denoted by '+'). However, Trail bikes also feature heavily in Cluster #3 along with most of the Enduro bikes.

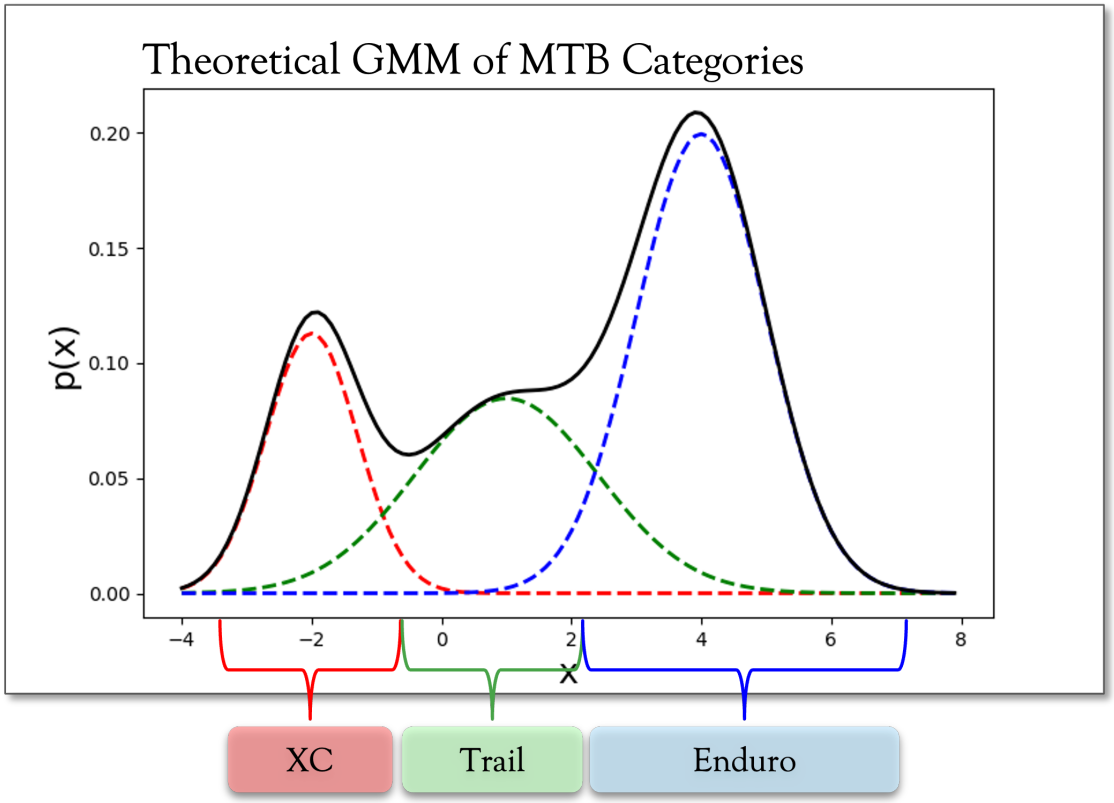
Overall, it's clear that there is significant overlap between our Clusters, mainly along the Principal Component 1 axis; lending credence to the notion that our bikes can be differentiated along a single, continuous scale.

Note: In the bottom-right of the graph (`PC2 < -4`), we see two Niner bikes, almost acting as outliers. For a 5'10" rider Niner suggests a size Medium, which results in low reach numbers on its bikes. From the earlier PCA plot, we see that Reach heavily corresponds with PC2, and thus these bikes appear lower on the visual.

// Gaussian Mixture Model (GMM)

In this section, we'll take a more probabilistic model to our clustering. That is, we'll use a Guassian Mixture Model (GMM) to build out normally distributed subgroupings within our mountain bike dataset, where the densities of each of the subgroupings represents a probability that a bike belongs to that subgrouping. Unlike K-Means, which is a more centroid-based clustering method, GMM is more of a distribution-based clustering method.

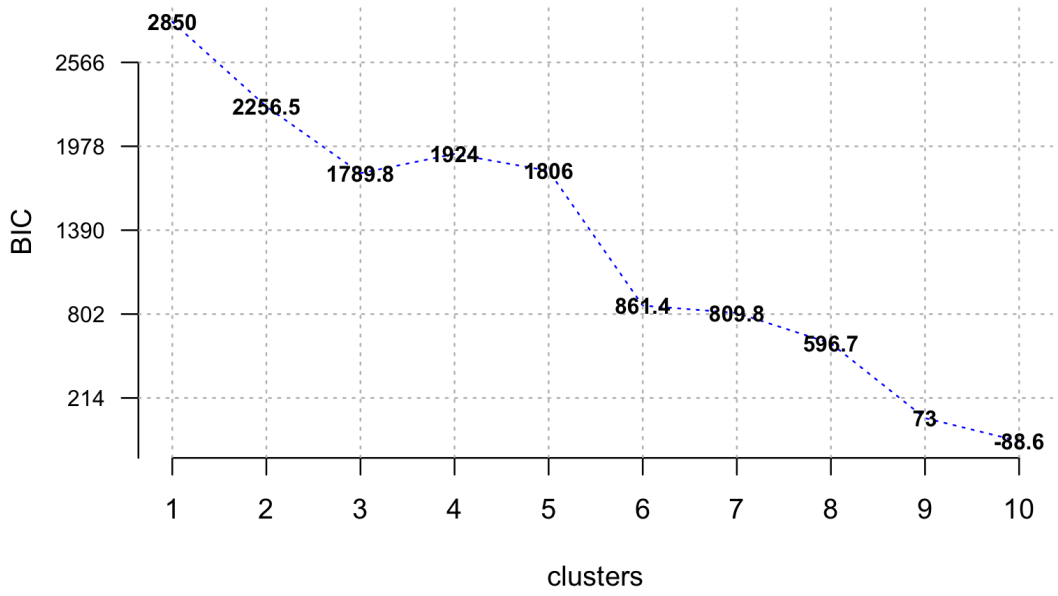
Generally, what we expect to see is something like the following:



where, given a specific type of bike, we can predict the probability, $p(x)$ that a bike belongs to a category like `Cross Country (xc)` vs. `Trail` vs. `Enduro`.

We'll run the `ClusterR::GMM()` function in R to figure out an optimal number of clusters. It uses the expectation-maximization algorithm to perform the probabilistic clustering; at each iteration, it aims to maximize the Bayesian Information Criterion (BIC) to determine an optimal number of clusters.

```
## iteration: 1  num-clusters: 1
## iteration: 2  num-clusters: 2
## iteration: 3  num-clusters: 3
## iteration: 4  num-clusters: 4
## iteration: 5  num-clusters: 5
## iteration: 6  num-clusters: 6
## iteration: 7  num-clusters: 7
## iteration: 8  num-clusters: 8
## iteration: 9  num-clusters: 9
## iteration: 10  num-clusters: 10
```



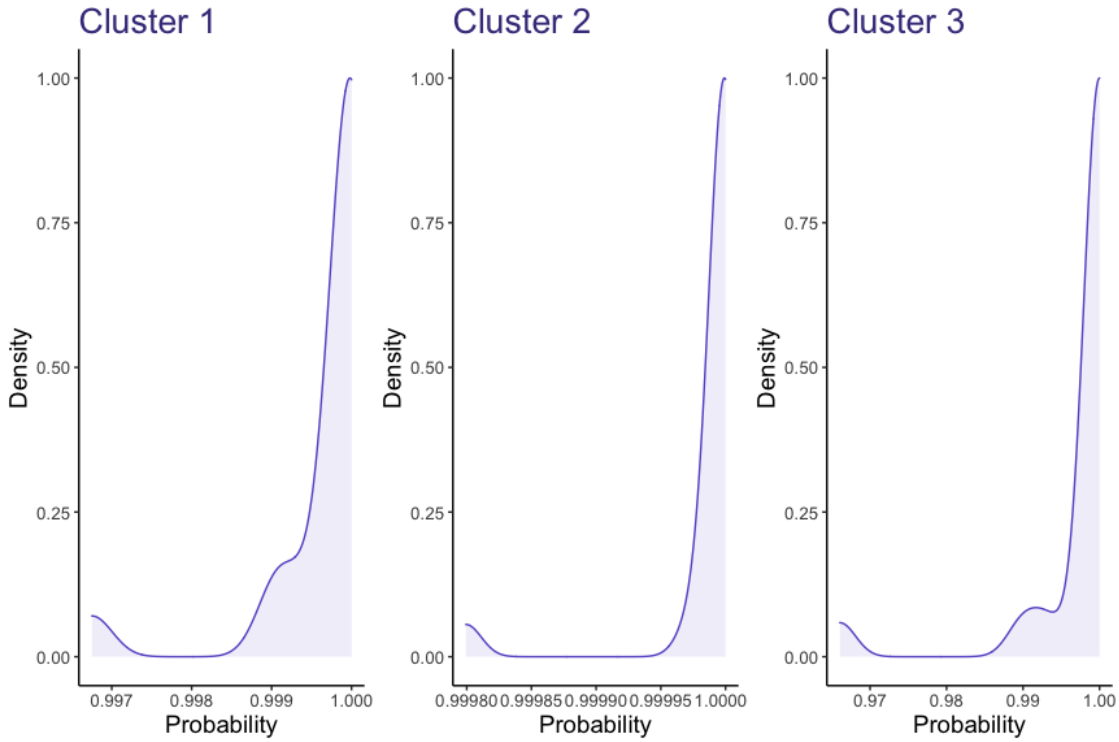
Optimal_GMM

From the plot above, we see that the BIC value decreases *generally* as the number of clusters increases. However, it appears that the first big drop is when `clusters = 3`. Let's try that value out and see which bike categories get mapped into each of the 3 clusters.

cluster_labels	am	dc	en	tr	xc
1	1	0	5	11	1
2	2	1	9	8	0
3	0	7	0	2	11

Here we see the predicted cluster labels along with the actual 5 bike categories in our data. Trail and Enduro bikes are mostly grouped into Clusters 1 and 2, while Downcountry and Cross Country bikes are grouped into Cluster 3. This would lead us to believe that the original 5 bike categories can be sufficiently explained with fewer clusters.

Even so, let's see how the probability of each bike belonging to a cluster appears by looking at the densities of each of the associated probabilities for a bike belonging to one of the cluster labels.



Above, we see the expected probability associated with predicting a correct class label. That is, the graph on the left shows how accurate the 3-cluster Gaussian Mixture Model was for predicting bikes fitting into Cluster #1. Generally, the probabilities for the predictions are all above .95; that is, GMM is extremely confident in grouping the different bikes into these 3 clusters.

// Multi-class SVM

If we took a slightly different approach and opted to treat our pre-assigned `label`s as truth, then we could approach this analysis as a supervised learning problem.

For this section, we chose to group the All Mountain and Enduro categories, since they completely overlapped in the PCA chart earlier. We also chose to switch the categorization of the Downcountry category, leaving it as a separate category and grouping it with both Trail and XC to experiment with the results of the model.

We chose to use a Multi-Class Support Vector Machine (SVM), and a grid search to tune the kernel functions and γ values. For each set of parameters, we used 10-fold cross validation on all rows of the data. We decided against holding out data as a test set since we have such limited data, and the 10-fold CV should evaluate the model's performance on blind data.

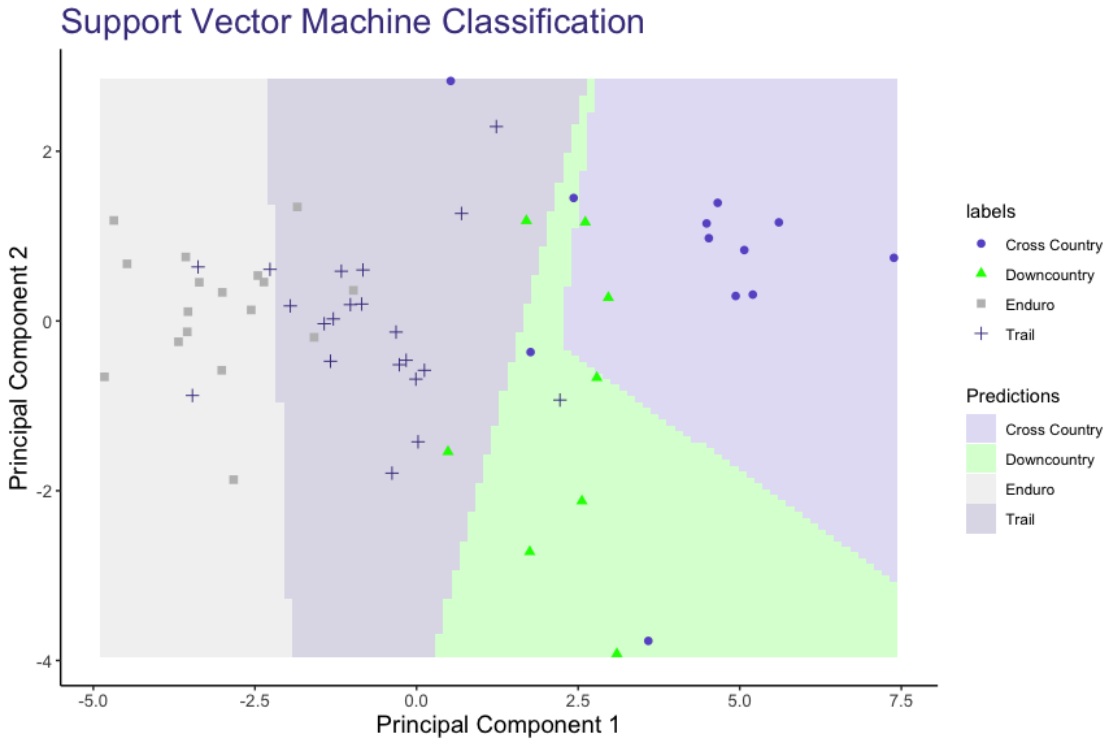
Using all of the data, the best SVM model was 73% accurate, using a Radial Basis kernel function with $\gamma = 2.595024$

Treating the Downcountry category as XC, the best model was 81.6% accurate, with a Radial Basis kernel function with $\gamma = 0.02983647$

Treating the Downcountry category as Trail, the best model was 80.0% accurate with a Radial Basis kernel function with $\gamma = 3.764936$

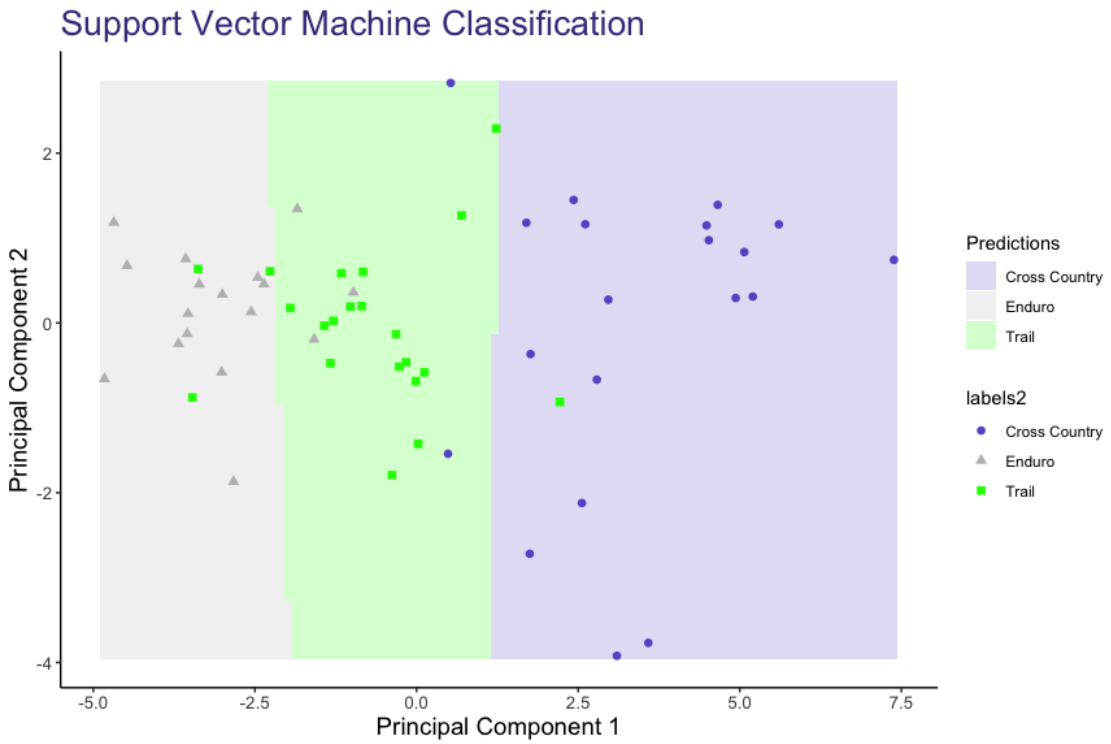
Logically, grouping Downcountry bikes with one of their adjacent categories should naturally lead to an increase in performance; however, this could suggest that the Downcountry category is slightly more skewed towards the XC bikes.

To visualize the results of the SVM classifier, we mapped all features to the 2 Principal Components. In so doing, we were able to achieve a higher accuracy of 75% using a linear kernel, while continuing to treat the Downcountry category as its own distinct category.



In the chart above, we can see how SVM closely maps the actual labels of the bike categories to their predicted categories. There are some misclassifications (when different colored dots appear in a differently shaded region), but overall the shaded SVM feature spaces seem to be accurate representations of the bike categories.

An interesting observation is that most of the boundary lines are more or less vertical, suggesting that most of the variation between classes is along Principal Component 1. We see this deviate between the XC and Downcountry boundary, however the validity of this boundary is still in question since the Downcountry category itself is more or less unofficial. Let's try remapping Downcountry bikes to the Cross Country (XC) category and review the updated results.



Mapping all Downcountry bikes to XC, the boundaries become almost entirely vertical, again suggesting that the classification of bikes can be attributed to Principal Component 1. The distinct regions appear to be accurate predictors of the 3 bike categories and are simple to explain, which would mean that we have effectively reduced the bias in our model.

/ Conclusions

// Findings

- All results suggest that trying to discretely categorize full suspension mountain bikes is more or less arbitrary.
- The categorization of a mountain bike should be treated as on a continuous scale, with Cross Country (XC) bikes on one end and Enduro (EN) bikes on another.
- To obtain where a specific bike lies on this scale, one can use the linear combination of the bike's specifications and the first principal component.
- This new spectrum of mapping bikes can provide bike manufacturers and consumers a method to quantify how a bike will handle when ridden.

Let's look at an example from the data above. Some bike companies, like Transition and Revel, do not explicitly categorize their bikes like others do. For these brands, we categorized them based on general attributes, as well as media coverage of them. Let's take a look at the Revel Ranger:

Revel_Ranger

Rear Travel	Fork Travel	Front Piston	Front Rotor Diameter	Rear Piston	Rear Rotor Diameter	Head Angle	Seat Angle	Crank Length	Stem Length	Handlebar Width	Reac
115	120	2	180	2	160	67.5	75.3	170	40	780	473

Converting to an input vector:

$$\text{Ranger} = \begin{bmatrix} 115 \\ 120 \\ 2 \\ 180 \\ 2 \\ 160 \\ 67.5 \\ 75.3 \\ 170 \\ 40 \\ 780 \\ 473 \\ 619 \\ 1194 \\ 436 \\ 338 \\ 699 \end{bmatrix} \rightarrow \text{Ranger (scaled)} = \begin{bmatrix} -0.61 \\ -0.80 \\ -1.68 \\ -0.53 \\ -1.68 \\ -1.30 \\ 0.79 \\ -0.39 \\ -0.40 \\ -1.15 \\ -0.11 \\ 0.60 \\ 0.16 \\ -0.33 \\ 0.16 \\ -0.16 \\ -1.01 \end{bmatrix} \text{ and PC1} = \begin{bmatrix} -0.31 \\ -0.31 \\ -0.23 \\ -0.26 \\ -0.23 \\ -0.27 \\ 0.30 \\ -0.26 \\ -0.01 \\ 0.24 \\ -0.27 \\ -0.17 \\ -0.26 \\ -0.30 \\ -0.13 \\ -0.25 \\ 0.08 \end{bmatrix}$$

$$\text{Ranger} \times \text{PC1} = 1.7$$

Mapping the Revel Ranger to its first Principal Component we get a value of 1.7 which puts us right around the boundary between Trail and XC, which lines up with the PinkBike editors' labelling of Downcountry in the video mentioned in the Project Overview section.

// Opportunities for Improved Analysis

There are a few opportunities to improve the analysis included in this presentation and forthcoming report:

- **Inclusion of more bikes (rows)** | The most obvious improvement we can make is to add more data points to our dataset. The data for each individual bike was manually entered by one of the authors of this report. After entering data for bikes of most major bike brands, we had enough data to accurately visualize the different bike categories; however, with more bikes, our algorithms will become more robust and less affected by the presence of outliers.
- **Inclusion of more bike features (columns)** | Although we included the most meaningful specs/geometry of the bikes analyzed, there are dozens of other, smaller features that can be used to help differentiate between different types of bikes.
- **Include all sizes of bikes** | We chose to use the size that corresponded to a 5'10" rider, but some bike manufacturers could interpret this as a Medium while others interpret this as a Large.
- **Include bikes across multiple years** | As bike trends slowly change, it'd be interesting to see how the data shifts across time. For example, the Rocky Mountain Element reduced its head angle from 70 degrees to 65.8 in one iteration of the bike. Including data from past years could provide valuable market insights into how the industry as a whole is moving.

// Lessons Learned

As both authors of this report are in the midst of the Online Masters of Science of Analytics (OMSA) program, we feel that it provided reinforcement on previously seen data mining topics as well as good introductions to net new topics. Here are some individual takes on the lessons learned from the course.

Mike: I think this course was a great complement to ISYE6740, Computational Data Analytics (CDA), which was a bit more theory focused, e.g. requiring us to build ML algorithms from scratch, while this course seemed more practical focused, e.g. HW 5 allowing us to use machine learning methods on any dataset of our choice. Additionally, I think this course was a good course to take concurrently with Data Visual Analytics, as it also required the use of Random Forests, and provided a good introduction to data analysis techniques that are standard across any project (e.g. cross validation).

Justin: I really appreciated the practical nature of this course. I've already had the opportunity to use some of the classification and machine learning algorithms at work. Although this is a different take on ISYE 6740, Computational Data Analytics (CDA), I appreciated getting to do a similar-style course in my programming language of choice (R) and to get to really dive into the actual code, which has already paid numerous dividends in a practical setting. I took this course concurrently with ISYE 6414, Regression Analysis. Although 6414 was recommended as a prerequisite for this course, it ended up being a nice pairing of courses.

// Course Suggestions

This course was a good introduction for practical applications. Some areas of improvement we would like to see included:

- Requiring more fundamental knowledge checks for some material. More specifically, we think the topics around Information Criterion and Discriminant analysis were glossed over too quickly. This course was our first exposure to those topics, so we're not sure if they were intentionally light on the material since they are not used in industry as much.
- Additionally, we think the assignments in this course could have been more clear, however, we also think this could have been intentional to mimic the ambiguity of working in industry.
- A lot of the quizzes ended up needing (or at least seriously benefiting from) pre-written code based on the knowledge checks. Writing the code was fun to make the quizzes more bearable; however, we would've loved to see the instructor/TAs post this kind of code after the quiz due dates so students could benefit from understanding how to solve the problems in R.