Senior Thesis in Mathematics

Theoretical Properties of Oversampling Techniques

Author:
Summer Will

Advisor:
Dr. Jo Hardin

Submitted to Pomona College in Partial Fulfillment
of the Degree of Bachelor of Arts

May 8, 2023

Abstract

Making predictions on imbalanced data is a challenge in many industries, including finance and medical research. Standard classification technologies typically bias prediction towards the majority class, resulting in poor accuracy for minority cases. Often, as in situations like credit card fraud or cancerous gene prediction, the minority cases are the ones statisticians are most keen on understanding. The advancement of gene expression technology and the proliferation of data availability has prompted statisticians to engineer ways to wrangle data in order to better make predictions on the minority class. Standard practice is to combine undersampling and oversampling. The purpose of this paper is to exposit a method of oversampling that goes beyond the standard practice of duplicating existing points–synthetic minority oversampling technique. We describe Chawla's 2002 algorithm and discuss the theoretical properties of SMOTE-augmented data sets, as well as classifier performance on high-dimensional data sets [Chawla et al., 2002].

Contents

Chapter 1

Background

1.1 Imbalanced Data

A framework for the problem of imbalanced data will be provided by discussing why such data leads us astray and how previous methods of fixing this problem have been insufficient. We will focus on the problem of making predictions on two classes only.

Balanced data occurs when a data set has roughly the same percentage of points for two different defining characteristics, or classes. For example, a randomly sampled data set that captures demographic info on a group of people can be divided into two classes: male and female. Imbalanced data occurs when the defining characteristic being assessed divides into two vastly uneven classes. For instance, if a researcher is interested in learning about the difference between fraudulent and normal credit card transactions, a data set that captures transactions in any given region or time period would have vastly more normal transactions because of how prolific credit card transactions generally are. Imbalance is prevalent in many fields such as internet, finance, and biomedical research. Imbalance becomes a problem when the characteristic that is most important to understand is the one that manifests as a minority.

1.2 Balancing Techniques

Initial practices for rectifying imbalanced data include undersampling and oversampling. Undersamping involves randomly removing as many points from the majority class as needed in order to make the two classes even. Undersampling is a popular starting point to fix imbalanced data and is sufficient for situations where the imbalance is not serious. Standard oversampling involves randomly duplicating existing minority points until the classes are roughly equivalent in size. This practice often does not accurately capture the behavior of the minority class, particularly in extreme cases of imbalance. The small number of points in the minority class means that any extreme value, or any value that does not reflect the natural behavior of a group (an outlier), has a great deal of relative gravity when duplicated during the oversampling process.

Standard oversampling by duplicating existing points compounds the effect of outliers. In another sense, the point of fixing imbalanced data is to try to more accurately represent the minority class, and duplicating points strengthens the existing insufficient representation.

1.3 Machine Learning Classification

Machine learning is the automated process of creating classification algorithms or functions on a set of data. Algorithms or functions that use information from a data set to make class predictions on that data are called classifiers. There are many different types of classifiers. Some form a divide between classes ranging in complexity from a line to hyperplane depending on the dimension of a data set. Others systematically make predictions based on Euclidean distance between points. Model-building requires a method of measuring performance that can motivate the creation of better models. Accuracy–the proportion of points classified correctly–is a common method of classifier evaluation. Imbalanced data requires a more sophisticated type of performance assessment, since a classifier trained to classify every incoming point in the majority class might have an accuracy as high as 99%, if the minority class represents only 1% of the data.

1.4 Performance Assessment

A Receiver Operating Characteristic curve (ROC curve) plots false positive rate versus true positive rate. A curve represents a single classifier and its performance under varying cutoffs, where higher area under the curve indicates better overall performance when compared to other classifiers (other curves).

1.5 K-Nearest Neighbors Classifier

K-Nearest Neighbors (k-NN) is a classification method whose algorithm is used within the SMOTE process. In order to motivate a discussion of the SMOTE algorithm, we include a description of k-NN [James et al., 2013] using notation from Blagus and Lusa [Blagus and Lusa, 2013]:

> Given a positive integer k and a test observation x_0, the k-NN classifier first identifies the k points in the training data that are closest to x_0, represented by R_0. It then estimates the conditional probability for class j as the fraction of points in R_0 whose response values equal j:
>
> $$\Pr(Y = j|X = x_0) = \frac{1}{k} \sum_{i \epsilon R_0} I(y_i = j). \text{ [James et al., 2013]}$$

In other words, given a data point x_0 whose class is unknown, select k nearest points–according to Euclidean distance–whose classes are known, then predict the class of x_0 based on the most common class out of the k neighbors. For example, if the five nearest points to some data point consist of three majority class points and two minority class points, then the point of interest will be classified as a majority class point (if the classification threshold is 0.5). A k-NN classifier uses a certain number of training data nearest to the test data predictor, then estimates the probability that the outcome will belong to a certain class.

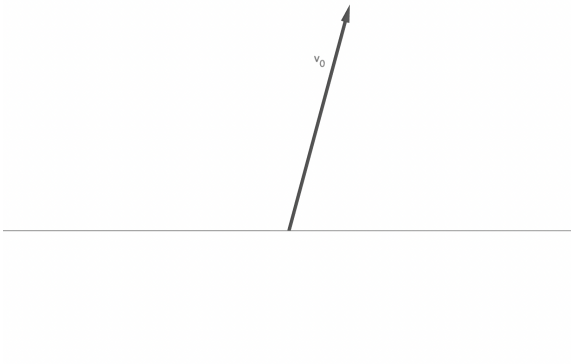
Chapter 2

SMOTE Algorithm

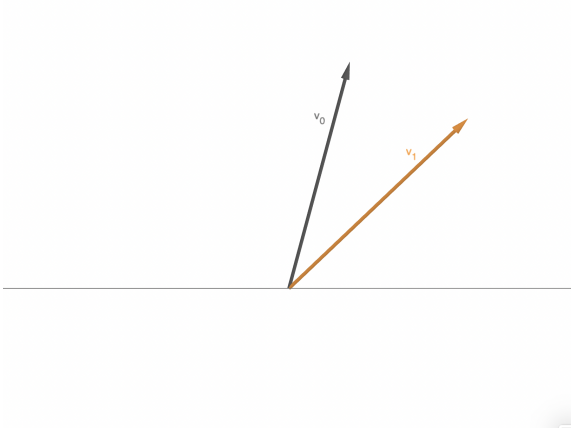
Standard oversampling techniques lack the ability to build a better understanding of class behavior; they typically magnify the impact of existing points. SMOTE enhances standard oversampling by creating synthetic points that deviate slightly from existing points but are still based off of them. While SMOTE does not add new information to the data set, by investigating the algorithm's theoretical properties we can discover which classifiers to use and which ones to avoid.

The SMOTE algorithm is a simple process that works as follows:

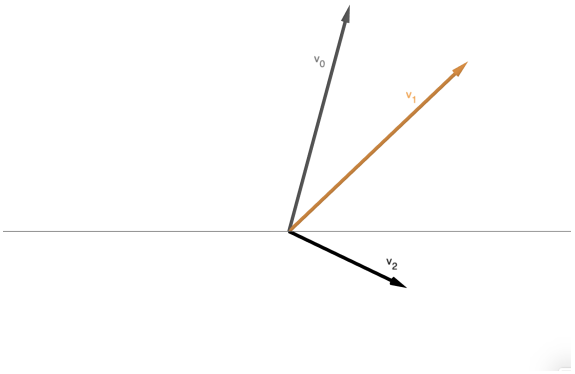
1. Select an existing (real) minority data point, or vector. Call it v_0.



2. Establish k, the number of nearest neighboring vectors to v_0. Out of the k nearest neighbors, select one at random, call it v_1.

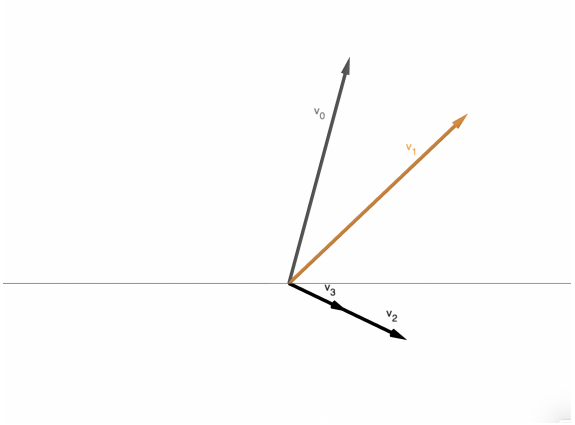


3. Subtract v_1 and v_0 to get v_2.



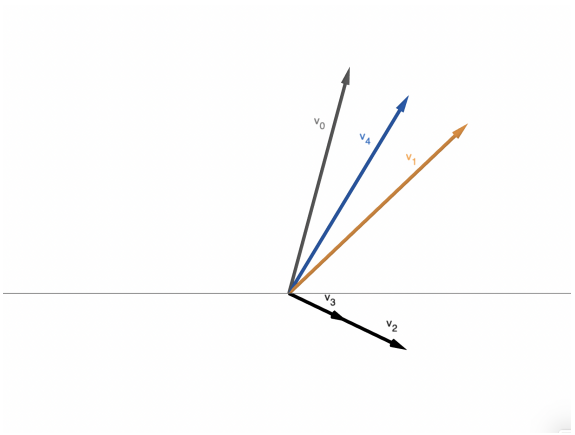
$$v_2 \quad = \quad v_1 - v_0$$

4. Multiply v_2 by a random scalar between 0 and 1, call the new vector v_3.



$$v_3 = c \cdot v_2$$

5. Add v_3 to the original data vector v_0 to get the final vector v_4.



$$v_4 = v_3 + v_0$$

v_4 is now a synthetically-created minority data point. The previously described process is the creation of 1 additional minority data point. Repeat

the process for every existing minority data point until balance is achieved.

When using SMOTE, standard practice is to combine undersampling of the majority and the synthetic oversampling of the minority to achieve desired balance. For example, say a dataset contained 10 minority observations and 250 majority ones. A researcher using SMOTE would first undersample by removing a number of the majority points and then synthetically oversample the minority. If the researcher decided to randomly remove 150 majority points, 9 cycles of the SMOTE algorithm would be done in order to achieve 50/50 balance. Specifically, each minority point would be used 9 times in the synthetic creation of a new points.

SMOTE utilizes randomness in its algorithm in order to best mimic the natural world. The prevalence of imbalanced data prompted statistical researchers to improve existing methods of rectifying imbalanced data. The natural world in theory involve points that are slightly different than existing minority data points. SMOTE does its best to mimic the existing data and thus the natural process it describes through strategic inclusion of randomness in the algorithm. Randomness comes from two steps in the algorithm: the random choice from k nearest neighbors and multiplying by a random number between 0 and 1. The randomness is reigned in by the relationship of new synthetic points to existing minority vectors.

Chapter 3

Theoretical Properties of SMOTE-Augmented Data

The SMOTE algorithm changes certain statistical and probabilistic properties of the data it augments, impacting classifier performance in intricate ways. This section provides comprehensive guides to the proofs behind theoretical changes to expected value, variance, correlation, and Euclidean distance in SMOTE-augmented data sets. Chapter 4 will discuss practical consequences of these changes.
The notation and steps from all proofs come directly from Blagus and Lusa. [Blagus and Lusa, 2013, 15]

3.1 Notation

Let $\mathbf{X} = \{X_1, X_2, ..., X_p\}$ be the p random variables measured for data point (or observation) $\mathbf{X}$. Define the jth dimension of a SMOTE sample ($\mathbf{S} = \{S_1, ..., S_p\}$), as

$$S_j = X_j + U(R_j - X_j),$$

where $\mathbf{X}$ is a sample from the minority class and $\mathbf{R} = \{R_1, ..., R_p\}$ is a randomly chosen sample among the k samples from the minority class with one of the k smallest Euclidean distances from sample $\mathbf{X}$; U is a uniform random variable defined on the interval $(0, 1)$, independent of the other variables. The subscripts indicate the variables of a sample.

3.2 Methods

For the high-dimensional simulations, Blagus and Lusa used samples of 100, with each observation having 1000 variables. 20 variables were chosen to deferentially express between classes. The variables were simulated from a normal distribution, with those in Class 1 having mean 0 and variance 1, where the variables of Class 2 were simulated with mean $\mu = 2, 1, .7, .5$, given at random, and variance 1. The variables' correlation was constructed in block-exchangeable form, where sets of 10 had pairwise correlation of $\rho = 0, .2, .5, .8$. Blocks were independent from each other. 80 observations were allocated to the training set, and 20 to the test set. The test set was balanced, but the training set was given imbalance anywhere between .05 and .95 percent of the points in Class 1.

For the low-dimensional simulations, the same construction was used but observations had 5 or 10 variables that were all differentially expressed and correlated with $\rho = .8$. In addition, the sample sizes were varied by $n = 40, 80, 200)$.

3.3 Motivating Simulation Results

Blagus and Lusa conducted a different set of simulations than described above in order to empirically investigate expected value, variance, and correlation between different sample types coming from different dimensions. Table **??** summarizes the results of those simulations using simulation averages. One entry of 0.66, for example, provides the variance of a SMOTE sample created from data in a normal distribution using a dimension of 10000, or a data set with 10000 variables.

They built the data sets by establishing true population means and variances as equal to 1. $\rho(\mathbf{S},\mathbf{X})$ denotes the correlation coefficient between the SMOTE sample and the original sample used to generate it. The exponential distribution is positively asymmetric while the normal and uniform distributions are symmetric.

The following table summarizes the simulations.
Table 1

	Normal					Exponential					Uniform				
p	2	10	100	1,000	10,000	2	10	100	1,000	10,000	2	10	100	1,000	10,000
$E(\mathbf{X^{NN}})$	1.01	1.00	1.00	1.00	1.00	0.88	0.90	0.90	0.96	0.99	1.01	0.99	1.00	1.00	1.00
$var(\mathbf{X^{NN}})$	0.71	0.7	0.83	0.94	0.98	0.59	0.69	0.67	0.84	0.94	0.88	0.83	0.93	0.97	0.99
$\rho(\mathbf{X^{NN}}, \mathbf{X})$	0.69	0.67	0.22	0.07	0.02	0.68	0.73	0.22	0.05	0.02	0.69	0.70	0.23	0.07	0.02
$E(\mathbf{R})$	1.04	1.00	1.00	1.00	1.00	0.79	0.87	0.91	0.97	0.99	1.02	0.99	1.00	1.00	1.00
$Var(\mathbf{R})$	0.62	0.70	0.86	0.95	0.98	0.45	0.62	0.69	0.87	0.95	0.82	0.86	0.94	0.98	0.99
$\rho(\mathbf{R}, \mathbf{X})$	0.36	0.56	0.17	0.05	0.02	0.28	0.60	0.16	0.04	0.01	0.34	0.59	0.19	0.06	0.02
$E(\mathbf{S})$	1.03	1.00	1.00	1.00	1.00	0.90	0.93	0.95	0.98	0.99	1.01	0.99	1.00	1.00	1.00
$var(\mathbf{S})$	0.67	0.73	0.68	0.66	0.66	0.58	0.70	0.60	0.64	0.66	0.79	0.80	0.70	0.68	0.67
$\rho(\mathbf{S}, \mathbf{X})$	0.74	0.86	0.71	0.66	0.63	0.67	0.87	0.73	0.65	0.63	0.69	0.87	0.70	0.66	0.64

Table 3.1: Each cell is an average of observation values in the simulation conducted by Blagus and Lusa. E denotes expected value, var variance, and ρ correlation. 2, 10, 100, 1,000, and 10,000 refer to number of variables for each observation in a sample. Observations were pulled from Normal, Exponential, and Uniform distributions.

The simulation results align with the authors' theoretical findings (See Chapter 4). Observe the expected value rows for $\mathbf{X}$ and $\mathbf{S}$ in any distribution, in the 10,000 variables column: the entries are around 1; expected value of a SMOTE sample closely matches the expected value of the original samples, both equalling 1. Observe the variance rows for $\mathbf{X}$ and $\mathbf{S}$ in any distribution, in the 10,000 variables column: the entries are .66; in high dimensions, variance of SMOTE samples is roughly 2/3 the variance of the original samples. Observe the row $\rho(\mathbf{S},\mathbf{X})$ in any distribution, in the 10,000 variables column: whereas $\rho(\mathbf{R},\mathbf{X})$ and $\rho(\mathbf{R},X^{NN})$ approach zero in high dimensions for any distribution, $\rho(\mathbf{S},\mathbf{X})$ is .63, .63, and .64, indicating SMOTE introduces a correlation between original and SMOTE samples.

The following sections will provide a theoretical framework for and corroborate these simulation findings.

3.4 Expected Value

The expected value of a class that has been augmented by SMOTE is the same as the expected value of the same class that has not been augmented by SMOTE.

In general, we wish to show $E(\mathbf{X}) = E(\mathbf{S})$. Samples themselves are vectors, in order to more easily prove theoretical properties Blagus and Lusa isolated samples into one variable, the jth.

Claim:

For high-dimensional samples or for samples taken from symmetric distributions:

$$E(S_j) = E(X_j) \tag{3.1}$$

Proof:

$$\begin{aligned} E(S_j) &= E(X_j + U(R_j - X_j)) \\ E(S_j) &= E(X_j) + E(U)(E(R_j) - E(X_j)) \\ E(S_j) &= E(X_j) + E(U)E(R_j) - E(U)E(X_j) \\ E(S_j) &= \frac{1}{2}(E(R_j) + E(X_j)) \end{aligned} \tag{3.2}$$

Recall from Section 3.1 that U is a uniform random variable defined on the interval $(0, 1)$. The final equation holds since $E(U)$ is equal to $\frac{1}{2}$. We are able to assign this value to $E(U)$ because "U is independent of the variables X_j and R_j, and because we assumed the equality of the expected values in the minority class." [Blagus and Lusa, 2013] Recall that the expected value of the SMOTE samples is equal to the expected value of the original samples for symmetric or high-dimensional data. So the final expression can be simplified to $E(S_j) = E(X_j)$.

It follows that $E(\mathbf{S}) = \frac{1}{2}(E(\mathbf{R}) + E(\mathbf{X}))$.

3.5 Variance

SMOTE decreases the variability of the (SMOTE-augmented) minority class

Claim:

$$\begin{aligned} var(S_j) &= \frac{1}{3}var(X_j) + \frac{1}{3}var(R_j) + \frac{1}{3}cov(X_j, R_j) + \\ & \frac{1}{12}E(X_j)^2 + \frac{1}{12}E(R_j)^2 - \frac{2}{12}E(X_j)E(R_j) \end{aligned} \tag{3.3}$$

Since $E(X_j) = E(R_j)$ for symmetric distributions, Equation 3.3 simplifies to

$$var(S_j) = \frac{1}{3}var(X_j) + \frac{1}{3}var(R_j) + \frac{1}{3}cov(X_j, R_j)$$

for high dimensional daa with covariance going to zero,

$$var(X_j) = var(R_j)$$

See Table 3.3.

Proof:

Recall:

$$\begin{aligned} E(S_j) &= E(X_j + U(R_j - X_j)) \\ E(U) &= \frac{1}{2} \\ E(U^2) &= \frac{1}{3} \end{aligned}$$

for $U(0, 1)$.

Variance can be expressed as:

$$var(S_j) = E(S_j^2) - E(S_j)^2 \tag{3.4}$$

Expand the first term:

$$\begin{aligned} E(S_j^2) &= E((X_j + U(R_j - X_j))^2) \\ &= E(X_j^2 + R_jUX_j - UX_j^2 + R_jUX_j + R_j^2U^2 - \\ & R_jU^2X_j - UX_j^2 - R_jU^2X_j + U^2X_j^2) \end{aligned}$$

Apply expected value to each term:

$$\begin{aligned} E(S_j^2) &= E(X_j^2) + E(R_j)E(U)E(X_j) - E(U)E(X_j^2) + \\ &\quad E(R_j)E(U)E(X_j) + E(R_j^2)E(U^2) - E(R_j)E(U^2)E(X_j) - \\ &\quad E(U)E(X_j^2) - E(R_j)E(U^2)E(X_j) + E(U^2)E(X_j^2) \end{aligned}$$

Replace each $E(U)$ and $E(U^2)$ with $\frac{1}{2}$ and $\frac{1}{3}$, respectively:

$$\begin{aligned} E(S_j^2) &= E(X_j^2) + \frac{1}{2}E(R_j)E(X_j) - \frac{1}{2}E(X_j^2) + \\ &\quad \frac{1}{2}E(R_j)E(X_j) + \frac{1}{3}E(R_j^2) - \frac{1}{3}E(R_j)E(X_j) - \\ &\quad \frac{1}{2}E(X_j^2) - \frac{1}{3}E(R_j)E(X_j) + \frac{1}{3}E(X_j^2) \end{aligned}$$

Group terms:

$$\begin{aligned} E(S_j^2) &= E(X_j^2) - \frac{1}{2}E(X_j^2) - \frac{1}{2}E(X_j^2) + \\ &\quad \frac{1}{3}E(X_j^2) + \frac{1}{2}E(R_j)E(X_j) + \frac{1}{2}E(R_j)E(X_j) \\ &\quad -\frac{1}{3}E(R_j)E(X_j^2) - \frac{1}{3}E(R_j)E(X_j) + \frac{1}{3}E(R_j^2) \end{aligned}$$

Simplify:

$$E(S_j^2) = \frac{1}{3}E(X_j^2) + \frac{1}{3}E(R_j)E(X_j) + \frac{1}{3}E(R_j^2)$$

We return to the latter part of equation (3.4). Recall:

$$E(\mathbf{S}) = \frac{1}{2}(E(\mathbf{R}) + E(\mathbf{X}))$$

So,

$$E(S_j)^2 = (\frac{1}{2}(E(\mathbf{R}) + E(\mathbf{X})))^2$$

Expand:

$$E(S_j)^2 = \frac{1}{4}E(R_j)^2 + \frac{1}{2}E(R_j)E(X_j) + \frac{1}{4}E(X_j)^2$$

Combine expressions from Equation (3.4):

$$\begin{aligned} var(S_j) &= \frac{1}{3}E(X_j^2) + \frac{1}{3}E(R_j)E(X_j) + \frac{1}{3}E(R_j^2) - \\ &\quad (\frac{1}{4}E(R_j)^2 + \frac{1}{2}E(R_j)E(X_j) + \frac{1}{4}E(X_j)^2) \end{aligned}$$

Split the $-\frac{1}{4}E(R^2)$ term into $-\frac{1}{3}E(R)^2 + \frac{1}{12}E(R)^2$, the $-\frac{1}{2}E(R_j)E(X_j)$ term into $-\frac{1}{3}E(X)E(R)$ and $-\frac{1}{6}E(X)E(R)$, the $\frac{1}{4}E(X_j)^2)$ term into $-\frac{1}{3}E(X)^2 + \frac{1}{12}E(X)^2$ to get the following:

$$\begin{aligned} var(S_j) &= \frac{1}{3}E(X^2) + \frac{1}{3}E(R^2) + \frac{1}{3}E(XR) \\ &\quad -\frac{1}{3}E(R)^2 + \frac{1}{12}E(R)^2 - \frac{1}{3}E(X)E(R) \\ &\quad -\frac{1}{6}E(X)E(R) - \frac{1}{3}E(X)^2 + \frac{1}{12}E(X)^2 \end{aligned}$$

Group terms:

$$\begin{aligned} var(S_j) &= \frac{1}{3}E(X^2) - \frac{1}{3}E(X)^2 + \frac{1}{12}E(X)^2 \\ &\quad +\frac{1}{3}E(R^2) + \frac{1}{12}E(R)^2 + \frac{1}{3}E(XR) \\ &\quad -\frac{1}{3}E(X)E(R) - \frac{2}{12}E(X)E(R) \end{aligned}$$

Group terms in such a way that we can use the definition of covariance in the last step:

$$\begin{aligned} var(S_j) &= \frac{1}{3}(E(X^2) - E(X)^2) + \frac{1}{3}(E(R^2) \\ &\quad -E(R)^2) + \frac{1}{3}(E(XR) - E(X)E(R)) \\ &\quad +\frac{1}{12}E(X)^2 + \frac{1}{12}E(R)^2 - \frac{2}{12}E(X)E(R) \end{aligned} \tag{3.5}$$

The last three terms of Equation 3.5 equal $\frac{1}{2}((E(X) - E(R))^2)$.

Use the definitions of variance and covariance to achieve the final result:

$$\begin{aligned} var(S_j) &= \frac{1}{3}var(X_j) + \frac{1}{3}var(R_j) + \frac{1}{3}cov(X_j, R_j) \\ &\quad + \frac{1}{12}E(X_j)^2 + \frac{1}{12}E(R_j)^2 - \frac{2}{12}E(R_j)E(X_j) \end{aligned}$$

$$\begin{aligned} var(S_j) &= \frac{1}{3}var(X_j) + \frac{1}{3}var(R_j) + \frac{1}{3}cov(X_j, R_j) \\ &\quad + \frac{1}{12}E(X_j)^2 + \frac{1}{12}E(R_j)^2 - \frac{2}{12}E(R_j)E(X_j) \end{aligned}$$

The last three terms of Equation 3.5 equal $\frac{1}{2}((E(X) - E(R))^2)$. In high dimensions, therefore, the simulations in Table 3.3 showed that the expected value of an original sample $\mathbf{X}$ approaches the expected value of a neighbor $\mathbf{R}$. Simulations showed that the correlation and thus the covariance of $\mathbf{R}$ and $\mathbf{X}$ is roughly 0 in high dimensions. So the above expression simplifies to:

$$\begin{aligned} var(S_j) &= \frac{1}{3}var(X_j) + \frac{1}{3}var(R_j) \\ &= \frac{2}{3}var(X_j) \end{aligned} \tag{3.6}$$

In Section 4.2 we will show the ramifications of a shrunken variance as a result of SMOTE.

3.6 Independence

SMOTE introduces a correlation between samples
Let S_j^s and S_j^t $s \neq t$ be the jth variables of two different SMOTE samples, defined as $S_j^s = X_j^s + U^s(R_j^s - X_j^s)$ and $S_j^t = X_j^t + U^t(R_j^t - X_j^t)$, where U^s and U^t are independent uniform variables $U(0, 1)$ and X_j^s, X_j^t, R_j^s and R_j^t are samples from the minority class. R_j^s and R_j^t are randomly chosen among the 5 nearest neighbors of X^s and X^t, respectively. The SMOTE algorithm could be modified to have a random choice among 3 or 7 nearest neighbors rather than 5, but 5 is conventional when using k-NN in general. [Blagus and Lusa, 2013]

S_j^s and S_j^t are still such that $s \neq t$ even if X^s and X^t or R^s and R^t that are involved in $X_j^s + U^s(R_j^s - X_j^s)$ and $X_j^t + U^t(R_j^t - X_j^t)$ might be the same. For clarity, depending on how imbalanced a dataset is, the SMOTE algorithm may be performed multiple times on each data point. So there are synthetically created points that are borne from the same original minority point. In addition, it is possible that two synthetic points are created using the same neighbor. Both of these things can happen at once as well. The following proof shows that if an original data point and/or its nearest neighbor are used for multiple SMOTE data points, a correlation is introduced.

We assume that $var(X_j^s) = var(X_j)$ for all the samples s of the minority class. [Blagus and Lusa, 2013]

Claim:

Assuming that the samples of the minority class are independent between each other and have the same variances for the jth variable $var(X_j^s) = var(X_j)$ for all the sth and tth samples of the minority class, the correlation (ρ) of the jth variable between SMOTE samples is

$$\rho(S_j^s, S_j^t) = \begin{cases} (\frac{1}{4}(var(X_j) + var(R_j)) + \frac{1}{2}cov(R_j, X_j))/var(S_j), & \text{if } (X_j^s = X_j^t \text{and} R_j^s = R_j^t) \text{or} (X_j^s, R_j^t \text{and} X_j^s = R_j^t) \\ (\frac{1}{4}(var(X_j) + \frac{1}{2}cov(R_j, X_j))/var(S_j), & \text{if } (X_j^s = X_j^t \text{and} R_j^s \neq R_j^t) \\ (\frac{1}{4}(var(X_j) + \frac{1}{2}cov(R_j, X_j))/var(S_j), & \text{if } (X_j^s \neq X_j^t \text{and} R_j^s = R_j^t) \\ 0, & \text{otherwise} \end{cases}$$

For high dimensions the correlation simplifies to:

$$\rho(S_j^s, S_j^t) = \begin{cases} \frac{3}{4}, & \text{if } (X_j^s = X_j^t \text{and} R_j^s = R_j^t) \text{or} (X_j^s, R_j^t \text{and} X_j^s = R_j^t) \\ \frac{3}{8}, & \text{if } (X_j^s = X_j^t \text{and} R_j^s \neq R_j^t) \\ \frac{3}{8}, & \text{if } (X_j^s \neq X_j^t \text{and} R_j^s = R_j^t) \\ 0, & \text{otherwise} \end{cases}$$

Proof:

$$\begin{aligned} cov(S_j^s, S_j^t) &= cov(X_j^s + U^s(R_j^s - X_j^s), X_j^t + U^t(R_j^t - X_j^t)) \\ &= cov(X_j^s, X_j^t) + E(U)cov(X_j^s, R_j^t) - E(U)cov(X_j^s, X_j^t) + \\ &\quad E(U)cov(R_j^s, X_j^t) + E(U)^2 cov(R_j^s, R_j^t) - E(U)^2 cov(R_j^s, X_j^t) - \\ &\quad E(U)cov(X_j^s, X_j^t) - E(U)^2 cov(X_j^s, R_j^t) + E(U)^2 cov(X_j^s, X_j^t) \\ &= \frac{1}{4}(cov(X_j^s, X_j^t) + cov(X_j^s, R_j^t) + cov(R_j^s, X_j^t) + cov(R_j^s, R_j^t) \quad (3.7) \end{aligned}$$

Equation 3.7 provides a general equation for the covariance between two

different SMOTE samples. In the process of creating a SMOTE sample, an original minority sample is randomly chosen and one of its nearest neighbors is randomly chosen, given by $\mathbf{X}$ and $\mathbf{R}$. Even if two SMOTE samples are not exactly the same it is possible for them to have been created using the same original sample, the same nearest neighbor, or both. The following piecewise function uses properties of the possibility of the same original or SMOTE sample so simplify Equation 3.7 in certain ways.

Assuming that the samples in the minority class are independent but can be correlated with their nearest neighbors we obtain

$$cov(S_j^s, S_j^t) = \begin{cases} \frac{1}{4}(var(X_j) + var(R_j)) + & \text{if } (X_j^s = X_j^t \text{and} R_j^s = R_j^t) \\ \quad \frac{1}{2}cov(R_j, X_j), & \quad \text{or}(X_j^s, R_j^t \text{and} X_j^s = R_j^t) \\ \frac{1}{4}(var(X_j)) + \frac{1}{2}cov(R_j, X_j), & \text{if } (X_j^s = X_j^t \text{and} R_j^s \neq R_j^t) \\ \frac{1}{4}(var(R_j)) + \frac{1}{2}cov(R_j, X_j), & \text{if } (X_j^s \neq X_j^t \text{and} R_j^s = R_j^t) \\ 0, & \text{otherwise} \end{cases}$$

For high-dimensional data, where covariance between original SMOTE samples and their nearest neighbors becomes zero, the result simplifies to

$$cov(S_j^s, S_j^t) = \begin{cases} \frac{1}{2}var(X_j), & \text{if } (X_j^s = X_j^t \text{and} R_j^s = R_j^t) \text{or}(X_j^s, R_j^t \text{and} X_j^s = R_j^t) \\ \frac{1}{4}var(X_j), & \text{if } (X_j^s = X_j^t \text{and} R_j^s \neq R_j^t) \\ \frac{1}{4}var(X_j), & \text{if } (X_j^s \neq X_j^t \text{and} R_j^s = R_j^t) \\ 0, & \text{otherwise} \end{cases}$$

The correlations between two variables are derived with the usual formula

$$\rho(X, Y) = \frac{cov(X,Y)}{\sqrt{var(X)var(Y)}}$$

Using this formula we arrive at the claim:

$$\rho(S_j^s, S_j^t) = \begin{cases} (\frac{1}{4}(var(X_j) + var(R_j)) + \frac{1}{2}cov(R_j, X_j))/var(S_j), & \text{if } (X_j^s = X_j^t \text{and} R_j^s = R_j^t) \text{or}(X_j^s, R_j^t \text{and} X_j^s = R_j^t) \\ (\frac{1}{4}(var(X_j) + \frac{1}{2}cov(R_j, X_j))/var(S_j), & \text{if } (X_j^s = X_j^t \text{and} R_j^s \neq R_j^t) \\ (\frac{1}{4}(var(X_j) + \frac{1}{2}cov(R_j, X_j))/var(S_j), & \text{if } (X_j^s \neq X_j^t \text{and} R_j^s = R_j^t) \\ 0, & \text{otherwise} \end{cases}$$

High dimensions allow us to assume that the a sample and a nearest neighbor can be treated as independent minority samples, leading to the following simplification:

$$\rho(S_j^s, S_j^t) = \begin{cases} \frac{3}{4}, & \text{if } (X_j^s = X_j^t \text{and} R_j^s = R_j^t) \text{or} (X_j^s, R_j^t \text{and} X_j^s = R_j^t) \\ \frac{3}{8}, & \text{if } (X_j^s = X_j^t \text{and} R_j^s \neq R_j^t) \\ \frac{3}{8}, & \text{if } (X_j^s \neq X_j^t \text{and} R_j^s = R_j^t) \\ 0, & \text{otherwise} \end{cases}$$

The variance of the SMOTE samples was found in Section 3.5. The covariance between a SMOTE sample and original samples in the high-dimensional setting can be derived using the same procedure described above, and is equal to

$$\rho(S_j^s, X_j) = \begin{cases} \frac{1}{2} var(X_j), & \text{if } (X_j^s = X_j \text{or} X_j = R_j^s) \\ 0, & \text{otherwise} \end{cases}$$

therefore the correlation between a SMOTE sample and an original sample is equal to $\frac{\sqrt{3}}{2\sqrt{2}}$.

In practice the correlation between a SMOTE and original sample and two SMOTE samples tend to be even higher because each original sample has a slightly positive correlation with its nearest neighbor.

We have shown that there is some correlation between a SMOTE an original sample in high dimensions.

3.7 Euclidean Distance

SMOTE reduces the expected Euclidean distance between test samples and the (SMOTE-augmented) minority class

Claim:

$$E(d^2(X^{test}, X)) = 2p \cdot var(X) > 2p\tfrac{5}{6} \cdot var(X) = E(d^2(X^{test}, S))$$

p represents the number of variables, so the addition of the $\frac{5}{6}$ term shows that under general circumstances, the squared Euclidean distance between a test sample and a SMOTE sample is less than that between a test sample and an original sample.

Proof:
Euclidean squared distance between a test and original sample is given below. The fact that a test sample and an original sample are independent

allows us to reach the following simplification. The expected value of a test sample and the expected value of an original sample are the same. The fact that $var(X) = E(X^2) - E(X)^2$ allows us, in particular, to make the last simplification:

$$\begin{aligned} E\left(d^2\left(\mathbf{X}^{\text{test}}, \mathbf{X}\right)\right) &= E\left(\sum_{j=1}^{p}\left(X_j^{\text{test}} - X_j^s\right)^2\right) \\ &= \sum_{j=1}^{p} E\left(\left(X_j^{\text{test}} - X_j^s\right)^2\right) \\ &= \sum_{j=1}^{p}\left(E\left(\left(X_j^{\text{test}}\right)^2\right) - 2E\left(X_j^{\text{test}}\right)E\left(X_j^s\right) + E\left(\left(X_j^s\right)^2\right)\right) \\ &= 2 \cdot p \cdot \text{var}(X) \end{aligned}$$

Below is the Euclidean squared distance between a test observation and a SMOTE sample.

$$\begin{aligned} E\left(d^2\left(\mathbf{X}^{\text{test}}, \mathbf{S}\right)\right) =& E\left(\sum_{j=1}^{p}\left(X_j^{\text{test}} - (X_j + U(R_j - X_j))\right)^2\right) \\ =& \sum_{j=1}^{p} E\left(\left(X_j^{\text{test}} - (X_j + U(R_j - X_j))\right)^2\right) \\ =& \sum_{j=1}^{p} E((X_j^{\text{test}} - X_j - UR_j + UX_j)^2) \\ =& \sum_{j=1}^{p}\left(\frac{4}{3}E\left(X_j^2\right) + \frac{1}{3}E\left(R_j^2\right) + \frac{1}{3}E\left(X_j R_j\right) - E\left(X_j\right)^2 - E\left(X_j\right)E\left(R_j\right)\right) \\ =& p \cdot \left(\frac{4}{3}\text{var}\left(X_j\right) + \frac{1}{3}\text{var}\left(R_j\right) + \frac{1}{3}\text{cov}\left(X_j, R_j\right) + \right. \\ & \left. + \frac{1}{3}E\left(X_j\right)^2 + \frac{1}{3}E\left(R_j\right)^2 - \frac{2}{3}E\left(X_j\right)E\left(R_j^{NN}\right)\right), \end{aligned}$$

For symmetric distributions the expression simplifies to

$$E\left(d^2\left(\mathbf{X}^{\text{test}}, \mathbf{S}\right)\right) = p \cdot \left(\frac{4}{3}\text{var}\left(X_j\right) + \frac{1}{3}\text{var}\left(R_j\right) + \frac{1}{3}\text{cov}\left(X_j, R_j\right)\right)$$

and for high-dimensional data to

$$E\left(d^2\left(\mathbf{X}^{\text{test}}, \mathbf{S}\right)\right) \;=\; 2 \cdot p \cdot \frac{5}{6} \cdot \text{var}(X)$$

We have shown that the expected value of Euclidean squared distance between a test sample and an original sample is less than that between a test sample and a SMOTE sample.

Chapter 4

Practical Consequences of Theoretical Properties

We have explored the theoretical properties of SMOTE-augmented data. In short: expected value of a data set is unchanged, variance is shrunk, correlation is introduced in certain contexts, and Euclidean distance is shortened in certain contexts. These theoretical changes have varying levels of impact on the particular classifier chosen to predict on the SMOTE-augmented data. The following sections will explore several examples of classifiers that are impacted in some way by SMOTE-augmented data and the mathematical background for why they are impacted.

Notation
G: general label for class number.
X: general label for a sample vector.
Σ: covariance matrix for a dataset.
Σ_k: covariance matrix for class k.
k: specific label for class number.
μ_k: mean of class k.

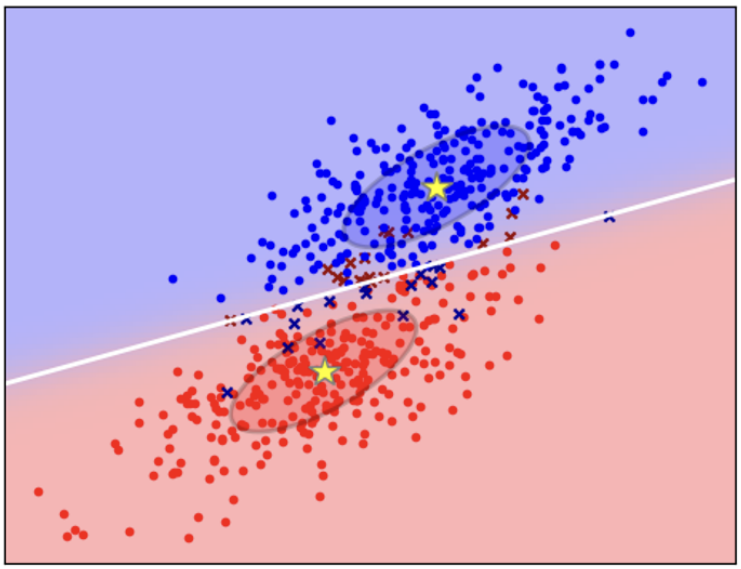


Figure 4.1: Linear Discriminant Analysis uses a linear boundary to divide and predict classes [Buitinck et al., 2013]

4.1 Expected Value: LDA and PAM

Linear Discriminant Analysis

Linear discriminant analysis (LDA) is a classification algorithm used to find a linear combination of features that separates two or more classes of objects or events. It is depicted in Figure 4.1. [Buitinck et al., 2013]

Observe that the classifier built a line inside the data set that best separates two classes.

Each observation will be given a probability of class membership. The mathematics behind LDA classification, in conjunction with a certain theoretical property of SMOTE, will show why the LDA classifier does not perform differently depending on whether it is being used on SMOTE-augmented data.

Based on the class balance, each class $k \in \{1, \ldots, K\}$ is assigned a prior $\hat{\pi}_k$ such that $\sum_{i=1}^{K} \hat{\pi}_k = 1$.

According to Bayes' rule, the posterior probability that object G belongs to class k is

$$\Pr(\hat{\mathrm{G}} = \mathrm{k}|\mathrm{X} = \mathrm{x}) = \frac{\mathrm{f_k(x)}\pi_\mathrm{k}}{\sum_{\mathrm{m}=1}^{\mathrm{K}} \mathrm{f_m(x)}\pi_\mathrm{m}}$$

where $f_k(x)$ is the density of X for class k. The posterior probability is a type of conditional probability that results from updating the prior probability with information summarized by the likelihood of G associated with a class k given G's observed data of x.

To estimate the most likely class given x:

$$G(x) = \arg\max_k \Pr(\mathrm{G} = \mathrm{k}|\mathrm{X} = \mathrm{x}) = \arg\max_\mathrm{k} \mathrm{f_k(x)}\pi_\mathrm{k} \tag{4.1}$$

LDA assumes the data likelihood is Gaussian:

$$f_k(x) = |2\pi\Sigma_k|^{-1/2} \exp\left(-\tfrac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k)\right)$$

Natural log is monotonically increasing, so maximum and minimum arguments of a function remain the same after the natural log of the function has been taken.

We plug the likelihood into the earlier classification function, Equation 4.1:

$$= \arg\max_k \delta_k(x) = \arg\max_k \log(f_k(x)\pi_k)$$

where

$$\delta_k(x) = -\tfrac{1}{2}\log|\Sigma| - \tfrac{1}{2}(x-\mu_k)^T\Sigma^{-1}(x-\mu_k) + \log\pi_k$$

and Σ is the covariance matrix of all the explanatory variables in the data set, regardless of class.

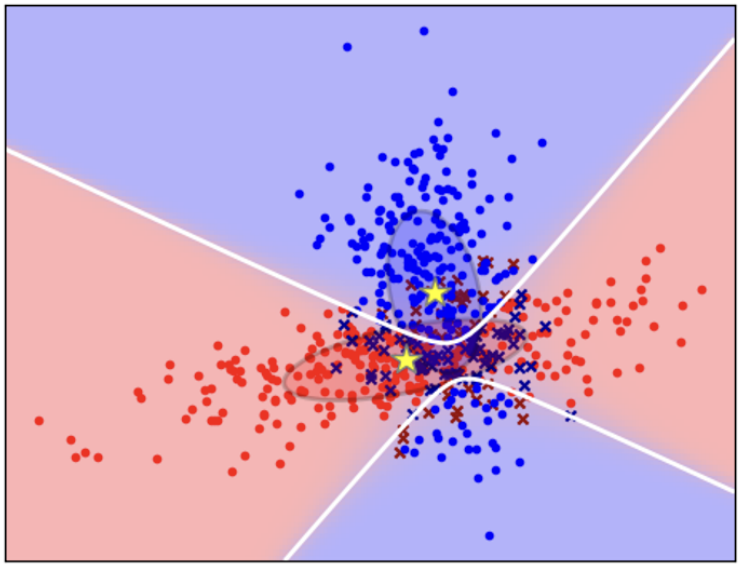
$$\hat{\Sigma} = \tfrac{1}{N-K}\sum_{k=1}^{K}\sum_{i\in k}(x_i-\hat{\mu}_k)(x_i-\hat{\mu}_k)^T$$

SMOTE is used specifically to improve the performance of classifiers and other prediction algorithms, yet the performance of LDA is unchanged on SMOTE-augmented data. In Chapter 3 we proved that SMOTE does not change the expected value of data, and we have shown how this impacts classifiers using the math behind LDA.

The following topic explores the probabilistic background of Quadratic Discriminant Analysis, a classifier that will behave differently on SMOTE-augmented data.

Quadratic Discriminant Analysis

Quadratic Discriminant Analysis is similar to LDA, except the borders it creates between classes is quadratic rather than linear, or is allotted two dimensions of flexibility. Observe Figure 4.2.



4.2: Quadratic Discriminant Analysis forms boundaries between classes using two dimensions and class-specific variances [Blagus and Lusa, 2013]

QDA behaves almost exactly like LDA, except it estimates $\sum_k$ separately for each class:

$$\delta_k(x) = -\tfrac{1}{2}\log|\Sigma_k| - \tfrac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k) + \log\pi_k\,.$$

$\delta_k(x)$ is plugged into Equation 4.1 in order to classify observations.
The classifier for SMOTE-augmented data must be chosen carefully; classifiers that rely heavily on class-specific variance, like QDA, are negatively impacted by SMOTE. In section 3.5 we discussed how applying the SMOTE algorithm to a data set impacts that data set's variance by shrinking it to $\frac{2}{3}$ its original size. Since SMOTE causes the data to have a variance that does not accurately reflect that natural phenomenon's variance in the real world, using classifiers that utilize variance is unwise for SMOTE-augmented data. Observe the size of the ellipses in Figures 4.1 and 4.2. LDA uses an overall variance, where the size of the ellipses are fixed between the two classes. QDA, however, as shown in the probabilistic interpretation, captures the varying orientation and shape of the ellipses among different classes. In data sets with differing variance structures between the two classes, QDA performs better than LDA because it captures the differing structures. However, see Section 3.5 for the proof that SMOTE decreases variability in the minority class, so fitting QDA onto SMOTE-augmented data empirically results in poor classifier performance. Due to its use of overall covariance rather than class-specific covariance, fitting LDA onto SMOTE-augmented data would result in performance that is equivalent before and after augmentation. QDA uses *class-specific variance*, which means that the classifier would be using the variance in the new, SMOTE-augmented and shrunken-variance minority class. This would cause a substantial change in the model between the non- and SMOTE-augmented data sets.

One classifier used by Blagus and Lusa whose performance is related to its unchanging expected value is Prediction Analysis for Microarrays (PAM)[Tibshirani et al., 2003], which is a classifier that uses centroids of clusters to determine a class. So from a mean-stand point it performs similarly to LDA and isn't affected that much by SMOTE. It differs from LDA particularly because it places more emphasis on prior probabilities, which take into account the existing proportion per class. Specifically, if a data set is balanced such that it consists of 50 percent one class and 50 percent the other, the prior probability for class 1 and class 2 are the same, each equal to .5. But if a data set is imbalanced at 99 to 1, the prior probability for class 1 is .99, for class 2 is .01.

The main issue with classifiers on imbalanced data is that they favor the majority class. The PAM classifier uses prior probabilities in its prediction, creating a bias towards the majority class when data is imbalanced. SMOTE

balances data, which nullifies the prior term in PAM, causing the classifier to improve with SMOTE and slightly outperform LDA, even though the classifiers are similarly affected by unchanged expected value. Both PAM and LDA use class proportions in their predictions. LDA uses priors–the original proportion of each class that makes up a data set–while PAM uses class-specific sample sizes. This discrepancy could be an explanation for why LDA's performance as a classifier is unaffected by SMOTE while PAM has a slight improvement in performance.

4.2 Variance: Quadratic Discriminant Analysis and Support Vector Machines

A Support Vector Machine (SVM) is a supervised learning model that constructs a boundary between classes and classifies new points according to their orientation to the boundary. SVMs behave similarly to QDA since both use boundaries in order to classify points and both operate in more than one dimension. It is not recommended, therefore, that QDA or SVM be used with SMOTE because SMOTE causes the variance of the minority class to shrink.

4.3 Correlation: Discriminant Analysis Methods, Penalized Logistic Regression, and Variable Selection

Logistic regression models probability onto a log-linear relationship between variables, and classifies points based on a probabilistic threshold. Penalized logistic regression penalizes for the number of variables added to a model to provide a mechanism against over-fitting, especially when the number of observations is less than the number of variables. Logistic regression assumes independence among samples [Blagus and Lusa, 2013], meaning that SMOTE might induce a correlation between observations.

4.4 Euclidean Distance: k-Nearest Neighbors in High Dimensions

In section 3.7, we discussed how SMOTE shrinks Euclidean distance between unclassified samples and synthetic data points. As dimension increases, the effect of this change in distance compounds, particularly for classifiers that rely heavily on Euclidean distance. One such classifier is k-NN, the only classifier to perform well on SMOTE-augmented data in high dimensions.

4.5 Simulation Results

Blagus and Lusa simulated the classifiers given in Table 4.5 and tested them in high and low dimensions. The green and red colors are used to indicate whether classifiers performed well or poorly. CO refers to "cut-off", or the classification threshold, which can be adjusted according to whether a researcher wants to make more or less Type I errors. NC means no correction– the classifiers were placed on imbalanced data where no method was used to fix the imbalance. An arrow up in the SMOTE columns means SMOTE improved classifier performance compared to the uncorrected data. An arrow down in the SMOTE columns means SMOTE worsened classifier performance compared to the uncorrected data, and roughly equal means the performance was roughly the same.

There were several occurrences that agreed with the above theoretical realizations. All the uncorrected classifiers assigned most of the test samples to the majority class. LDA, the classifier based on mean values, changed negligibly after being applied to SMOTE-augmented data. QDA, the classifier based on variance, was harmed by SMOTE. Similarly, SVM performed slightly worse after SMOTE. The most promising result is that SMOTE had a positive effect on k-NN classifiers. The predictive accuracy of the minority and majority classes on their own were approximately equal.

	low-dimensional data		**high-dimensional data, without variable selection**			**high-dimensional data, with variable selection**		
Classifier	NC	SMOTE	NC	CO	SMOTE	NC	CO	SMOTE
1-NN		↑		≈	↓		≈	↑
5-NN		↑		↑	↓		↑	↑
DLDA		≈		≈	≈		≈	≈
DQDA		≈		≈	↓		≈	↓
RF		↑		↑	↑		↑	≈
SVM		↑		↑	≈		↑	≈
PAM		↑		↑	≈		↑	↑
PLR-L1		↑		↑	≈		↑	≈
PLR-L2		↑		↑	≈		↑	≈
CART		↑		≈	≈		≈	≈

Table 4.5: Summary of results from Blagus and Lusa findings [Blagus and Lusa, 2013]

Chapter 5

Conclusion

The theoretical properties of Synthetic Minority Oversampling Technique include unchanging expected value, shrunken variance, introduced correlation between certain samples, and a smaller Euclidean distance between samples in some contexts. These theoretical properties are an inherent result of the SMOTE algorithm. The creation of SMOTE was one of several strides made in the statistics community in an effort to rectify imbalanced data. The algorithm is an improvement on standard oversampling by introducing randomness at select parts of the algorithm in order to try and reflect the behavior of the data in the natural world. SMOTE is one of the most popular methods of handling imbalanced data, but its limitations are not extensively explored. The purpose of this paper was to elucidate SMOTE's limitations in order to make better decisions on classifier types applied to SMOTE-augmented data.

Acknowledgements

First and foremost I would like to thank my advisor, Professor Hardin, who has been an essential and driving force in my mathematics education at Pomona. I would also like to thank the Pomona mathematics department, in particular Professors Sarkis, Garcia, and Aguilar whose lessons will serve me for years to come. In addition I would like to thank Professor Zemel in the economics department, Professor Mistry at Claremont-McKenna and Professor Bachman at Pitzer for helping me along my statistics journey. Finally, I would like to thank my friend Nam Do, who will probably be the only person that reads this.

Bibliography

[Blagus and Lusa, 2013] Blagus, R. and Lusa, L. (2013). Smote for high-dimensional class-imbalanced data. *BMC Bioinformatics*, 14.

[Buitinck et al., 2013] Buitinck, L., Louppe, G., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., Niculae, V., Prettenhofer, P., Gramfort, A., Grobler, J., Layton, R., Vanderplas, J., Joly, A., Holt, B., and Varoquaux, G. (2013). API design for machine learning software: experiences from the scikit-learn project.

[Chawla et al., 2002] Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *J. Artif. Intell. Res.*, 16:321–357.

[James et al., 2013] James, G., Witten, D., Tibshirani, R., and Hastie, T. (2013). *Introduction to Statistical Learning*. Springer Texts in Statistics, New York, NY.

[Tibshirani et al., 2003] Tibshirani, R., Hastie, T., Narasimhan, B., and Chu, G. (2003). Class prediction by nearest shrunken centroids, with applications to dna microarrays. *Statistical Science*, 18(1):104–117.