Methods in Open Science

Data Engineering and Prep

Image from NASA: https://commons.wikimedia.org/wiki/File:Earth_Western_Hemisphere_transparent_background.png#filelinks

WV VIEW

AmericaView
Empowering Earth Observation Education
AmericaView.org - Est. 2003

Before you can perform analyses to answer specific questions, it is commonly needed to prepare or clean data as input to specific tasks. For example, statistical inference, linear regression, and predictive modeling with machine learning all have some requirements regarding the format and characteristics of input data. In this section, we will explore methods used to prepare data. This process is generally termed data cleaning or feature space engineering in the context of predictive modeling.

Although these topics are not necessarily exciting, they are extremely important. Data scientists generally spend a large proportion of their time preparing data. For example, I work a lot with map data, and most of the time, data preparation is needed to complete my projects.

Book Recommendation

Kuhn, M. and Johnson, K., 2019. *Feature engineering and selection: A practical approach for predictive models*. CRC Press.

http://www.feat.engineering/

https://www.routledge.com/Feature-Engineering-and-Selection-A-Practical-Approach-for-Predictive-Models/Kuhn-Johnson/p/book/9781032090856

2

This is a great text that described the practical application of performing feature space engineering and feature selection in the context of predictive modeling and machine learning. Note that we will revisit some of these topics again when we discuss predictive modeling.

Common Data Cleaning and Prep Tasks/Issues

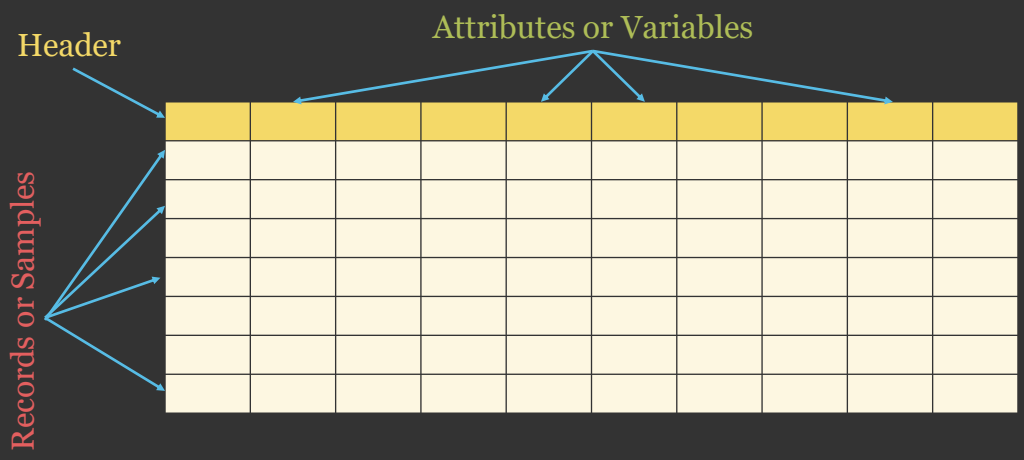
- Transpose
- Reshape
- Center
- Scale
- Range Scale
- Skewed or not Normally Distributed
- Outliers
- Leverage Samples
- Missing Data
- Recode/consolidate categories
- Big Data
- Convert numeric data to ordinal data
- Convert nominal data to dummy variables
- Date manipulation
- Limited number of samples
- Transform multiple variables

3

This slide lists some of the common data cleaning and preparation tasks undertaken by data scientists. We will discuss these topics one-by-one in the following slides.

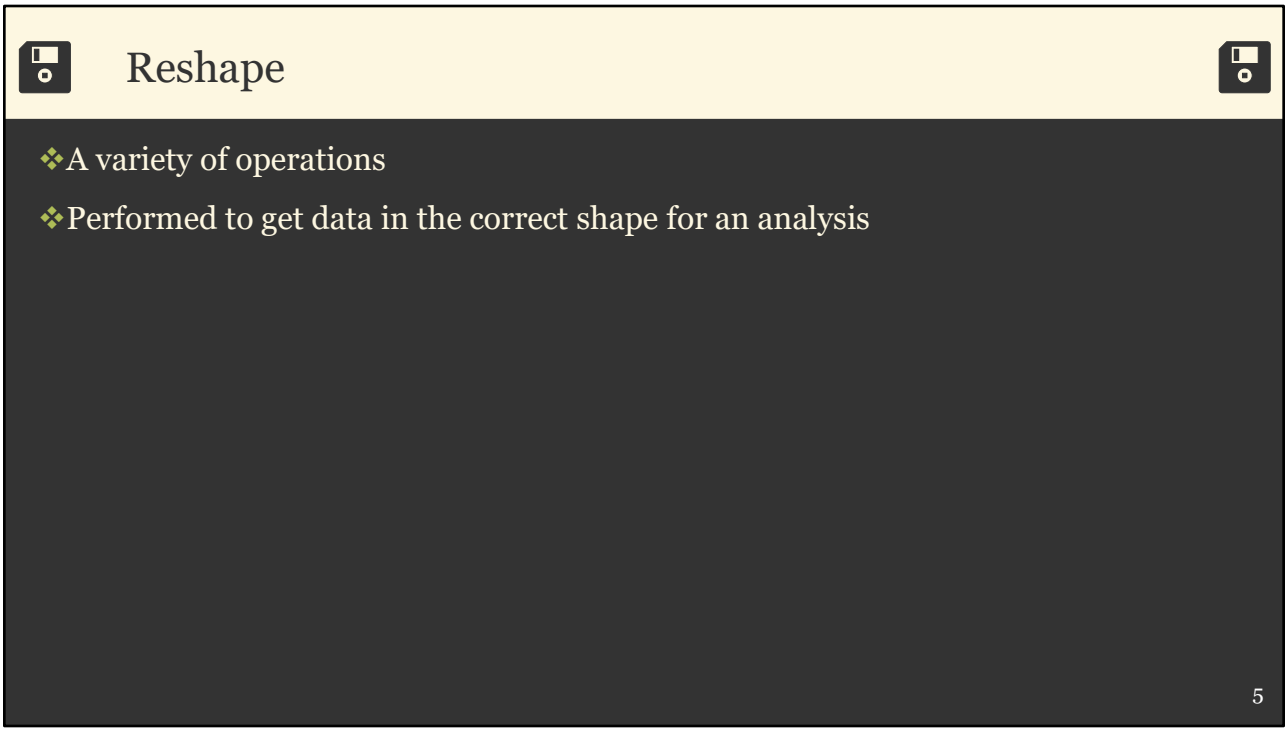
Transpose

- Rows → Columns
- Columns → Rows



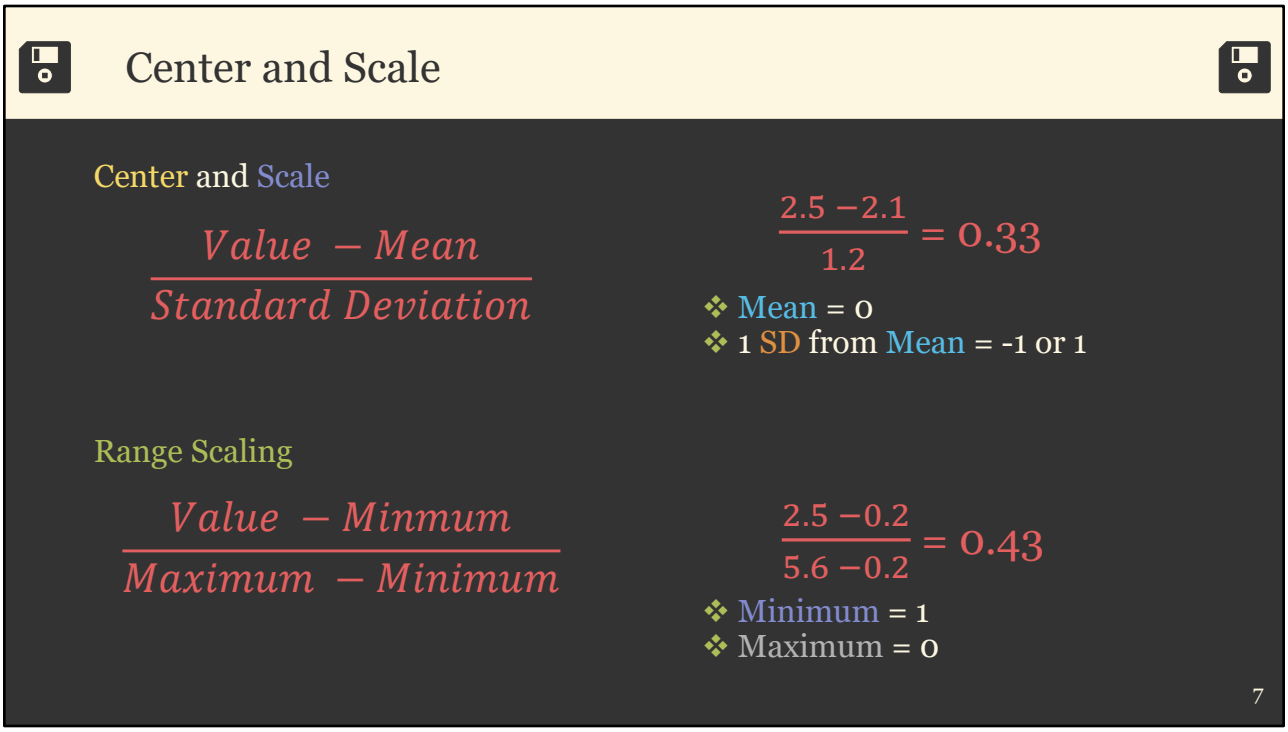
4

The most common form of data is to have each row correspond to a data point and each column represent the same piece of information about each data point. For example, each row could represent a student while each column could represent information about the student (age, academic year, GPA, etc.). However, some datasets may be formatted such that rows represent attributes while columns represent data points or samples. The process of transposing a table means to make the rows the columns the columns the rows. It may be necessary to transpose a table prior to performing an analysis.



Reshaping data is a bit more general than simply transposing a table. This could consist of a wide range of operations that are applied to get data into the correct shape for an analysis. For example, a table of temperature measurements may have mean temperatures per month separated into different columns. However, the analysis may require that all the measurements be aggregated into a single column with the month noted in a new column. All data science environments, including Python and R, have methods available to reshape and prepare data. This process can be complex depending on the structure of the original data and the format required for the analyses.

Methods for Single Variables



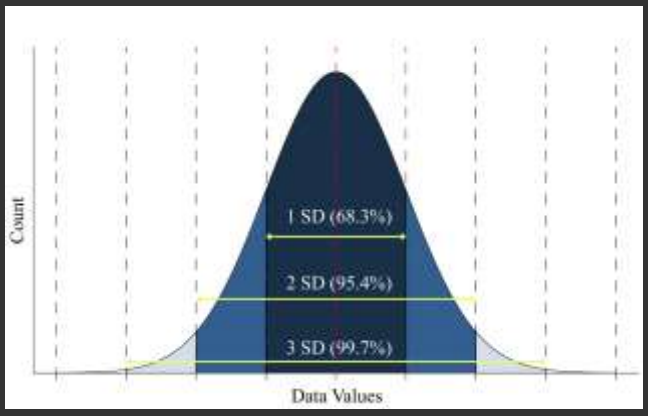
Some methods are sensitive to the scale of the data. For example, if variables are measured using different units or ranges of values, this can have a negative impact on calculations. In later modules, you will see examples of machine learning algorithms that require all variables to be on a consistent scale.

There are multiple ways to rescale data. The center and scale method consists of subtracting the mean from each data point (centering) then dividing by the standard deviation (scaling). Once this process is undertaken, the data will be centered at or have a mean of zero, and data points that are 1 standard deviation from the mean will have a value of -1 or 1.

Another option is range scaling in which the minimum is subtracted from each value and the result is divided by the range (maximum minus minimum). Using this method, the maximum value will scale to 1 and the lowest value will scale to 0.

The Normal Distribution

- Bell-shaped
- Mean, median, and mode are equal
- Curve is symmetrical about the mean
- Shape of the normal curve varies based on the mean and standard deviation of the data
- 68.3% of the data are within 1 standard deviation of the mean
- 95.4% of the data lie within 2 standard deviations of the mean
- 99.7% of the data lie within three standard deviations of the mean



8

Some methods require that input data follow or approximately follow a normal distribution. For example, this is a common requirement for some parametric tests and methods, such as linear regression. There are also other distributions that some statistical tests make use of, which we will talk about in the statistical inference module. Examples include the T-Distribution and F-Distribution. Here, we will focus on the normal distribution as an example.

The provided graphic conceptualizes a normal distribution. The x-axis represented the data values, and the y-axis represents the count or frequency of the values. This distribution has a bell-shape. Also, the mean, median, and mode are equivalent, and the curve is symmetrical about the mean. Since the mean, median, and mode are equivalent, then the curve is symmetrical about the mean, median, and mode.

In terms of the distribution, 68.3% of the data are within one standard deviation of the mean, 95.4% are within two standard deviations of the mean, and 99.7% are within three standard deviations of the mean. Note that a normal distribution can have different mean values and standard deviations. However, the data distribution relative to the mean and standard deviation should approximate these percentages.

No real data are perfectly normally distributed. However, some data are close

to normal or approximate this distribution. If data are not perfectly normally distributed, this is generally not an issue for most techniques. However, major deviations from normality can be an issue.

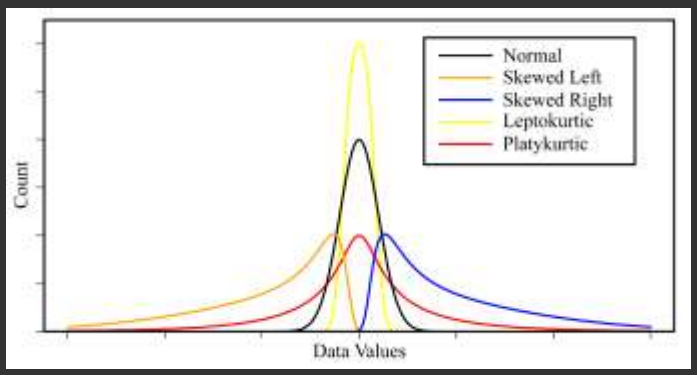
Skewness and Kurtosis

Skewness: measure of symmetry

- - = skew to left
- + = skew to right
- 0 = near normal

Kurtosis: heavy-tailed or light-tailed compared to a normal distribution

- - = light-tailed (leptokurtic)
- + = heavy-tailed (platykurtic)
- 0 = near normal (mesokurtic)



9

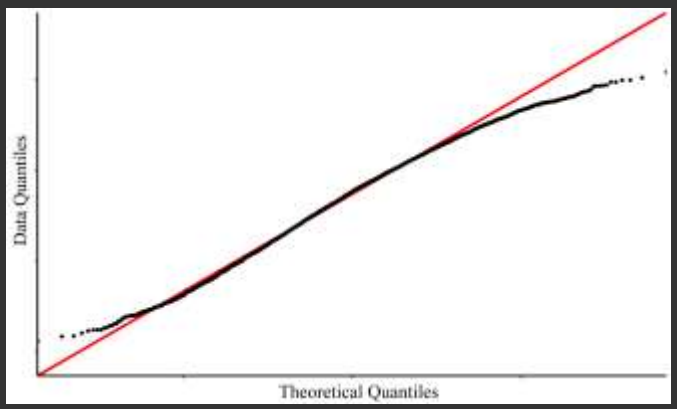
Skewness relates to divergence in symmetry about the mean. Data that are left skewed have a long tail to the left. In other words, they have some low values that are far from the mean. In contrast, data that are right-skewed have a tail to the right, or they have some high values that are far from the mean. I have always found this terminology a bit confusing, as I think of skewed data relative to the location of the most commonly occurring values. However, the terminology is in reference to the direction of the long tail, as described here.

Kurtosis is a measure of how variable the data are in comparison to a normal distribution. Kurtosis exists when the percent of the data within one, two, and three standard deviations from the mean vary from the values presented on the last slide. Data that are leptokurtic, or light-tailed, have values that are more concentrated close to the mean and are less variable than a normal distribution. This is represented by the yellow curve in the example plot. In contrast platykurtic distributions are heavy-tailed or are less concentrated close to the mean relative to a normal distribution. Mesokurtic indicates that the variability is close to normal.

Another issue worth mentioning is that data may be multimodal or have multiple “humps” or clusters in the curve. In such cases, means and medians may be misleading.

Quantile-Quantile Plot (Q-Q Plot)

- Compares data quantiles to theoretical quantities that represent normally distributed data



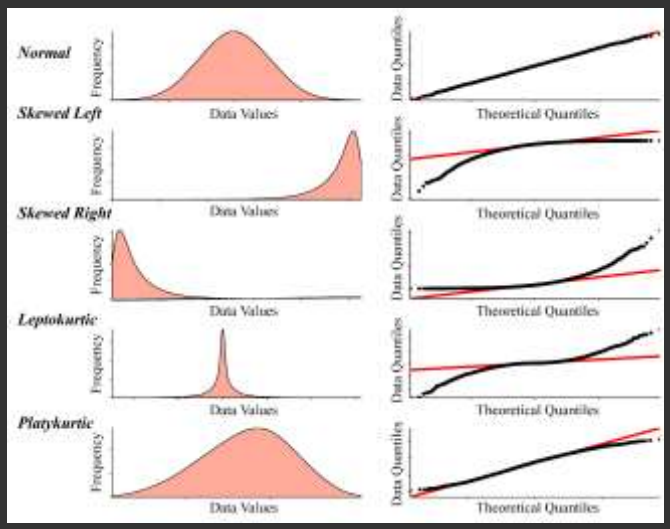
- Along reference line = normal distribution
- Off reference line = suggests issues in normality

10

A way to visualize or assess normality is by using a Q-Q Plot, or Quantile-Quantile Plot. A Q-Q Plot compares the quantiles of the data to theoretical quantiles that represent a normal distribution of the data.

Features that fall along the reference line have quantiles that approximate those of a normal distribution. If the points fall off of the reference line, this suggest issues in normality. The data may be skewed, have kurtosis issues, be multimodal, or have some combination of these issues.

Q-Q Plot



- On line = normal
- Concave = skewed to right
- Convex = skewed to the left
- High/low off line = kurtosis

11

Generally, Q-Q plots will take on a convex shape if the data are skewed to the left, a concave shape if skewed to the right, and will diverge from the reference line at high and low values if there are kurtosis issues. The graphic on this page provides some examples.

One issue with Q-Q Plots is that different users may not interpret them similarly. Extreme cases of skewness and kurtosis are generally obvious, but less severe cases will generally result in less obvious deviations from the reference line. In such situations, different users may come to different conclusions as to whether or not there is a normality issue.

Types of Data Transformations

- Natural log = compress ranges of larger values relative to smaller values
- Square root = compress ranges of larger values relative to smaller values (less extreme than natural log)
- Square = compress ranges of smaller values relative to larger values

12

A natural log or square root transformation can be used to compress ranges of larger values relative to smaller values. This could be useful when data are skewed to the right. In contrast, a square transformation compresses ranges of smaller values relative to larger values. This can be useful when data are skewed to the left.

Box-Cox Transformation

- Used to modify a skewed predictor variable so that the distribution is more normal
- Requires λ parameter
- Only works for positive values

$$x_{new} = \begin{cases} \frac{x_{old}^{\lambda} - 1}{\lambda} \; if \; \lambda \neq 0 \\ ln(x_{old}) \; if \; \lambda = 0 \end{cases}$$

Box, G.E. and Cox, D.R., 1964. An analysis of transformations. *Journal of the Royal Statistical Society: Series B (Methodological)*, *26*(2), pp.211-243.

13

It is sometimes possible to transform a variable so that its distribution more closely approximates a normal distribution. One method for doing this is the Box-Cox Transformation. This method requires the parameter lambda. If lambda is 1, no transformation is applied. Lambda values near zero apply a more natural log-style transformation while lambda values near 0.5 apply a more square root-style transformation. A lambda value of -1 takes the inverse of the values or divides one by the original values.

Yeo-Johnson Transformation

$$x_{new} = \begin{cases} \left((x_{old} + 1)^{\lambda} - 1\right)/\lambda & if\ \lambda \neq 0, x_{old} \geq 0 \\ ln(x_{old} + 1) & if\ \lambda = 0, x_{old} \geq 0 \\ -\left[(-x_{old} + 1)^{2-\lambda} - 1\right]/(2 - \lambda) & if\ \lambda \neq 2, x_{old} < 0 \\ -ln(-x_{old} + 1) & if\ \lambda = 2, x_{old} < 0 \end{cases}$$

Yeo, I.K. and Johnson, R.A., 2000. A new family of power transformations to improve normality or symmetry. *Biometrika*, *87*(4), pp.954-959.

14

The Box-Cox transformation only works for positive numbers. In contrast, the Yeo-Johnson transformation can be applied to negative, zero, and positive values.

Outlier Detection/Removal

What is the cause?

- Errors
- Not part of population
- Odd but real
- Subcomponent of population

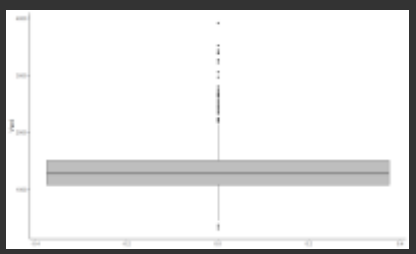
$$Z = \frac{X - \mu}{\sigma}$$

Finding Outliers

- Sort data
- Plot data
  - Box Plot
  - Histograms
- Z-scores
- Interquartile Range
  - 1.5 IQR above 3rd quartile
  - 1.5 IQR below 1st quartile
- Hypothesis tests

Solutions

- Remove sample
- Transform data
- Use different techniques



15

Outlier samples or data points generally have extremely large or extremely small values for one or more variables in comparison to the other samples in the dataset. Outliers can have large impacts when implementing some techniques and methods.

It is first important to consider the cause of outliers. If it can be shown that the data resulted from errors or are incorrect then it is generally okay to remove these samples without biasing the results. For example, if a temperature recording taken at a weather station is well outside of the normal temperatures at the site, or is physically impossible, it is generally safe to assume the data are in error.

You may also find that the some of the samples are not from the population you are interested in studying. For example, you may be collecting data associated with undergraduates but have accidently included some graduate students. In such cases, these samples can also generally be removed without issue.

However, sometimes high or low measurements represent real data, or you cannot say for certain that the data point(s) are in error. In such cases, it is generally recommended that these points be maintained as they do represent real data and removing them may skew the results or bias the analysis. In the

module associated with statistical inference, we will discuss the concept of creating an unbiased sample from a population.

By exploring your data, you may also find that there are distinct subpopulations within the larger population that have unique characteristics and show up as outliers. For example, some counties may have a high percentage of vacant homes because these counties have a lot of vacation properties. So, these counties are not in error, but they represent a unique subcomponent of the population with disparate percentages of vacant homes associated with a real reason.

There are different methods available to detect outliers. One of the simplest methods is to simply sort the data descending or ascending based on a column or attribute to find the largest and smallest values, respectively, and compare them to the nearest values in the sequence. This can also be visualized using box plots and/or histograms.

A more numeric method is to calculate Z-scores. Z-scores assume a normal data distribution and are calculated by subtracting the mean from the value and then dividing by the standard deviation. The mean has a Z-score of zero, and values that are one standard deviation form the mean have a Z-score of 1 or -1. If the Z-Score is more than 3, this indicates that the data point is more than three standard deviations from the mean. Another method is to use the interquartile range. High values that are more than 1.5 IQR from the 3rd quartile and low values that are lower than 1.5 IQR from the 1st quartile may be outliers. Remember that some software will automatically draw these samples in as points on a boxplot. For example, there are several high values that are more than 1.5 IQR from the 3rd quartile in the box plot included on this slide.

Lastly, there are statistical tests to assess, flag, or identify outliers.

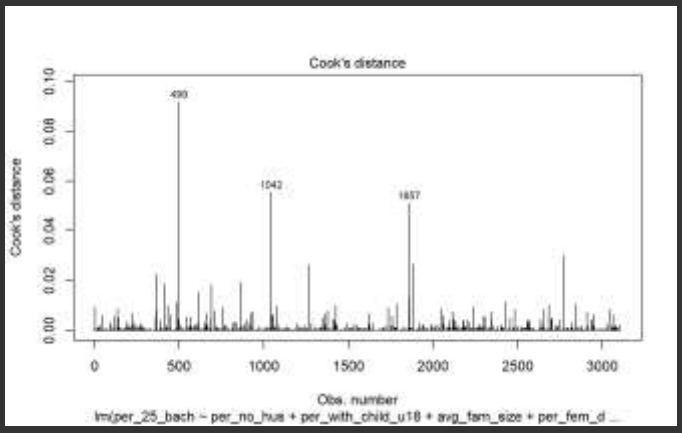
Note that all the methods just discussed that are used to find outliers only suggest that the values are very high or very low and may be substantially different from the other data points. These methods don't indicate the reason for these high or low values. In other words, just because a data point is flagged as an outlier using one or more of these methods, it doesn't mean that the data are in error.

There are a few solutions for dealing with outliers. One option is to remove these samples; however, this should only be done if the data points resulted from errors. If the data points are maintained in the dataset, transforming the data may minimize the impact of outliers. Lastly, you may decide that it is best

to use a technique or measure that is not very sensitive or less sensitive to outliers. For example, linear regression is generally sensitive to outliers whereas some machine learning methods, such as support vector machines, are less sensitive. As mentioned earlier, the mean is generally more impacted by outliers than the median. So, you may choose the median as a measure of central tendency as opposed to the mean when outliers are present but cannot be justifiably removed.

Leverage Samples

- Samples with unique combinations of values
- Can have large/disproportionate effect on some analyses
- Cook's Distance



Leverage samples or data points are commonly confused with outliers. A leverage point is one that has an odd combination of values. For example, most counties with a high median income may also have a high rate of college completion. However, there may be some counties that have a high median income and low college completion. So, the issue here is not extreme values for one or more of the variables, but odd combinations of the variables or attributes that are different from the general pattern represented in the other data points in the dataset.

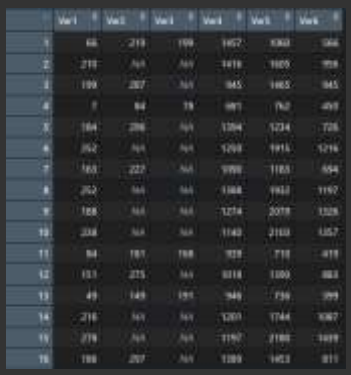
High leverage points are generally discussed in the context of regression as they tend to have a disproportionate impact on the final regression equation or model. These data points can be flagged using the Cook's Distance measure. How this measure works is outside the scope of this course.

Note that it might be worth investigating data points that have high leverage, as you may find that they represent a unique subgroup of the population or that these samples represent errors and should be removed from the analysis.

Missing Data

What is the cause?

- Random
- Not random



Finding Missing Data

- Gaps in table
- Assigned a unique code: NA, NaN, -9999, NULL, NODATA, etc.
- Can count number of missing records by row or column

Solutions

- Remove columns with missing data
- Remove rows with missing data
- Fill data gaps (interpolation)

17

A major issue when dealing with real datasets is that some measures may be missing for some data points. For example, a dataset representing information for all students attending a high school may have some features that are missing. Specifically, not every student may have an associated home address.

If missing data occur at random, then this may not bias the analysis since the sample is likely still representative of the population. Again, we will discuss the concept of unbiased sampling in the statistical inference module. However, if missing data are not random, this may introduce a bias in the analysis since the sample is no longer representative of the larger population.

Missing data may simply be empty fields in a table. In some tables and/or software tools specific codes indicate missing data. Examples include NA (not applicable), NaN (not a number), -9999, NULL, or NODATA. Some software may even use different missing codes for categorical vs. numeric data. When moving data between software tools or importing data, it is important that missing data are properly coded for the current working environment. Most statistical software packages have tools available to summarize missing data, such as counting the number of rows and/or columns with missing data.

The simplest solution for dealing with missing data is to delete any row or column with missing values. In a data table, the rows tend to represent

individual samples or data points whereas the columns represent a specific feature or variable. For example, in a dataset representing information about students in a high school, each record or row would be a student and each feature or column would be information about a student, such as the student's birth date. So, removing rows is effectively removing data points whereas removing columns is removing a specific variable from the analysis. There are several issues with this method. First, if there are a lot of missing data, then this may severely limit the number of samples available for the analysis. Further, if the missing data are not random, this could bias the analysis such that the sample is no longer representative of the population. For example, if students from a certain academic year are more likely to have missing records than students from other academic years, removing these records would cause the sample to underrepresent the students in that specific academic year or no longer be representative of the entire student population.

Another option, which is generally helpful when there are many missing values or the missing values are not random, is to estimate the missing values. This could be done by simply replacing the missing values with the mean or median of the available values. However, this would likely be biased if the missing values are not random. Another option is to estimate missing values from other data points that are likely to be similar to the missing data point. There are several methods to do this. One option is using the k-nearest neighbor machine learning methods, which we will discuss in the predictive modeling portion of the course. For time series data, you could fill gaps using the nearest values in the time series or values from a prior year and around the same day if you are trying to capture seasonal patterns.

In short, dealing with missing values can be complicated. Many methods are discipline and/or problem specific. If data points are removed or missing values are interpolated or estimated in some way, it is important that the methods are transparent and well documented to foster reproducibility.

Consolidate/Reclassify Categories or Numeric Values

Original	New
Aspen_Birch	Aspen_Birch
Douglas_Fir	Other
Eastern_Softwoods	Other
Elm_Ash_Cottonwood	Elm_Ash
Exotic_Hardwood	Other
Exotic_Softwood	Other
Fir_Spruce_Mountain_Hemlock	Spruce_Fir
Maple_Beech_Birch	Maple_Mix
Oak_Gum_Cypress	Other
Oak_Hickory	Oak_Dominant
Oak_Pine	Oak_Dominant
Other	Other
Other_Hardwood	Other
Spruce_Fir	Spruce_Fir
White_Red_Jack_Pine	Pine_Dominant

Original	New
0-25	1
25-50	2
50-75	3
75-100	4

18

For nominal data, you may need to recode or reclassify the current categories into a new set of categories. This is often undertaken when there are too many categories, or the categories are too detailed. In the example, I have taken a set of forest community types and consolidated them into a smaller number of categories.

You may also want to recode ranges of values for a numeric variable to an ordinal variable with a defined number of levels. This will involve determining what ranges of values you want to group. Note in the provided table that there is some ambiguity about what code the highest and lowest values in each bin will be assigned to in the new ordinal variable. When recoding ranges of values, “closed on the right and open on the left” implies that the highest value in the range will be included in that grouping whereas the lowest value will be included in the prior grouping. In contrast “open on the right and closed on the left” implies that the highest value will be included in the next grouping and the lowest value will be included in the current grouping.

Dummy Variables

Classes
B
C
A
A
C
A
D
A
B
B
D
D
C
A
C

A	B	C	D
0	1	0	0
0	0	1	0
1	0	0	0
1	0	0	0
0	0	1	0
1	0	0	0
0	0	0	1
1	0	0	0
0	1	0	0
0	1	0	0
0	0	0	1
0	0	0	1
0	0	1	0
1	0	0	0
0	0	1	0

19

Some methods are not able to accept nominal data as input. For example, some machine learning methods can only accept numeric predictor variables. As a result, we often need to convert nominal variables into a numeric representation. Although there are other ways to accomplish this, one of the most commonly used methods is to convert the single nominal attribute into a set of dummy variables.

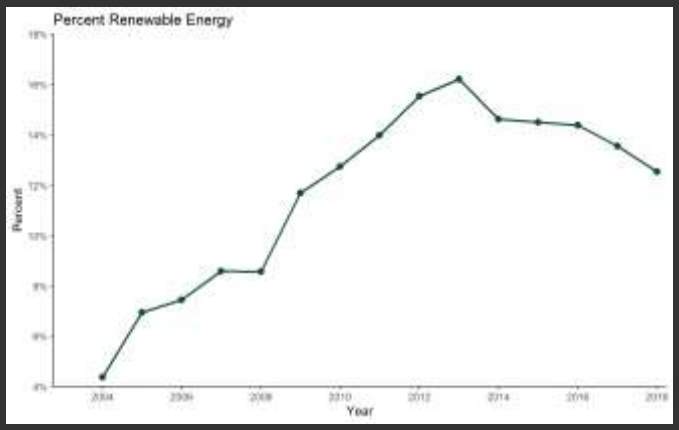
To create dummy variables, each level of the category, minus one category, becomes a new column. For a specific row, the column associated with the correct level receives a value of 1 while all other columns receive a value of 0. Effectively, the nominal data are now represented by multiple columns that can only hold 0 or 1.

One of the columns is dropped when creating the dummy variables. For example, we could drop the "D" column, which is shown in purple in the example. This is because if all other columns were coded to 0, then the "D" column is known to be 1. So, including this column does not provide any additional information. This can also result in an issue of perfectly correlated predictor variables, which can be a problem for some analyses.

If one category is not dropped, the encoding is generally referred to as one-hot encoding.

Time and Dates

- Character → Time/Date
- Number → Time/Date
- Standardize Time/Date
- Aggregate
  - Minutes → Hours
  - Hours → Days
  - Days → Months
  - Months → Seasons
  - Days → Years
  - Etc.
- Fill data gaps
- Smooth time series



20

Times and/or dates are generally treated as a unique data type and not just numbers and/or characters. This is because we commonly want to perform specific types of operations on these data, such as creating time series graphs, analyzing trends over time, or aggregating based on time steps. As a result, data science environments have defined data types for times/dates specifically.

One issue with times/dates is that there is not generally a standard way to represent them. For example, in the United States we commonly write the date as "Month Day Year" whereas is more common to use "Day Month Year" in other countries. Times make use of different time zones and can be reported with different levels of precision (seconds, minutes, hours, etc.). Thus, dates generally need to be converted into a common format prior to an analysis. Also, certain analyses may require the dates to be in a specific format.

You may also want to aggregate data to a different interval. For example, you could calculate the total precipitation over a month at a location by summing all the hourly or daily measurements. You could calculate the mean or median monthly temperature by aggregating daily measurements. Months could be aggregated to seasons or years.

It is also sometimes necessary to fill data gaps if an analysis requires measurements over a fixed, regular interval. Data gaps may result from sensor

malfunctions or inability to collect measurements. For example, a time series of satellite images are often irregularly spaced due to cloud cover.

Lastly, you may want to smooth the data to obtain a more general trend or remove noise.

Once time/date data are prepared, a variety of analyses can be performed, such as summarization, investigation of oscillating or seasonal patterns, long term trends, and forecasting into the future.

Transforming Multiple Variables

Transforming Multiple Variables

- Principal Component Analysis
- Kernel Principal Component Analysis
- Independent Component Analysis
- Canonical Correspondence Analysis
- Autoencoders
- Spatial Sign

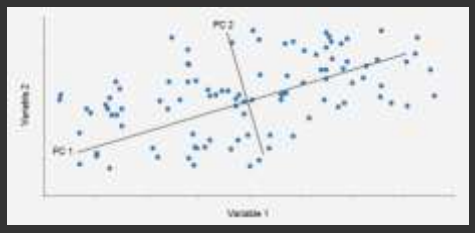
	Var1	Var2	Var3	Var4	Var5	Var6
1	66	219	199	1457	1060	566
2	210	357	350	1416	1605	956
3	199	287	330	945	1465	945
4	7	84	78	681	762	450
5	184	296	291	1394	1234	726
6	252	386	540	1250	1915	1216
7	163	227	279	1090	1183	694
8	252	379	561	1388	1932	1197
9	188	359	441	1274	2079	1326
10	238	383	524	1140	2103	1357
11	84	181	168	939	710	419
12	151	275	295	1018	1390	883
13	49	149	191	946	736	399
14	216	313	409	1201	1744	1087
15	278	433	634	1197	2180	1439
16	186	297	329	1389	1453	811

22

In this last section of the module, we will discuss methods for transforming multiple variables. This slide lists several techniques for transforming or combining multiple variables to create new variables. A full discussion of all available techniques is beyond the scope of this class. Instead, I will focus on principal component analysis as an example method to transform multiple variables to create new variables to include in an analysis. I will then then briefly mention some alternatives.

Principal Component Analysis (PCA)

- Orthogonal transformation
- Converts *n* correlated variables into *n* uncorrelated variables
- Variance is information
- Data variability is in a smaller number of principal components, reducing the number of needed variables



https://www.statistixl.com/features/principal-components/

23

One common issue with data is that the different features or variables tend to be correlated with each other. This results in redundant information based on the assumption that variability is information. So, it might be desirable to generate new, uncorrelated variables from the original features to capture the information content available in the original data in a smaller number of variables. This is the purpose of principal component analysis, or PCA.

Principal Component Analysis (PCA)

- **Eigenvalues** = Variance explained by each principal component
  - Larger Eigenvalue = More variance explained
- **Eigenvectors** = Values needed to create principal components from a linear combination of the input variables

EIGENVALUES AND EIGENVECTORS

Number of Input Layers: 6 — Number of Principal Component Layers: 6

PC Layer	1	2	3	4	5	6
Eigenvalues	1441846.83621	271831.53245	45726.35454	2928.82266	1993.02311	708.18719
Eigenvectors – Input Layer						
1	0.07643	0.31423	0.41331	0.53785	0.30618	0.07478
2	0.13114	0.31426	0.44427	0.50922	0.24305	-0.72937
3	0.15877	0.43231	0.43667	-0.38515	-0.08208	0.09658
4	0.79381	-0.54500	0.24239	-0.09595	0.05435	0.03264
5	0.48906	0.35524	-0.53907	0.50504	-0.29589	0.02840
6	0.28712	0.43763	-0.38578	-0.61413	0.58389	-0.03853

$$PC1 = (0.076*Var1) + (0.131*Var2) + (0.159*Var3) + (0.794*Var4) + (0.489*Var5) + (0.287*Var6)$$

24

The process of principal component analysis involved the transformation of the raw variables into a new space in which the resulting features, or principal components, are not correlated. Calculating PCA involves some matrix algebra and the use of a covariance matrix. A mathematical proof of this concept is beyond the scope of this course. The key concept here is that PCA allows for the determination of Eigenvectors, which are effectively coefficients that can be applied to each variable to obtain a new variable or principal component that explains a certain proportion of the variance in the raw input data. The relative amount of variance explained by a single principal component is associated with its Eigenvalue. Larger Eigenvalues indicate a larger proportion of variance explained.

For example, in the provided table 6 input variables are being used to obtain 6 uncorrelated, new principal component variables. To obtain the first principal component specifically, the following equation is applied:

$$PC1 = (V1 * 0.07643) + (V2 * 0.13114) + (V3 * 0.15877) + (V4 * 0.79381) + (V5 * 0.48906) + (V6 * 0.28712).$$

Other principal component variables can be calculated using their associated coefficients.

In summary, the process of PCA allows for the determination of Eigenvectors which are used to calculate principal components from a linear combination of the input variables.

Principal Component Analysis (PCA)

- Generally, the first principal component (PC) explains the most variability in the data
- Explained variability decreases with increased PC
- Can visualize variance with a Scree Plot

PERCENT AND ACCUMULATIVE EIGENVALUES

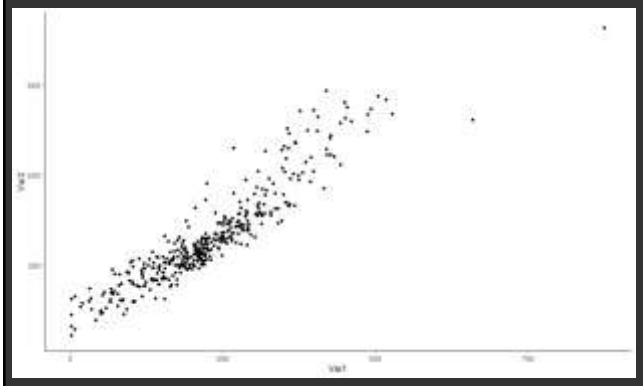
PC Layer	EigenValue	Percent of EigenValues	Accumulative of EigenValues
1	1441868.03621	81.6987	81.6987
2	271631.53245	15.3913	97.0900
3	45726.35454	2.5910	99.6810
4	2926.82286	0.1658	99.8469
5	1983.02311	0.1124	99.9593
6	798.18719	0.0481	100.0000

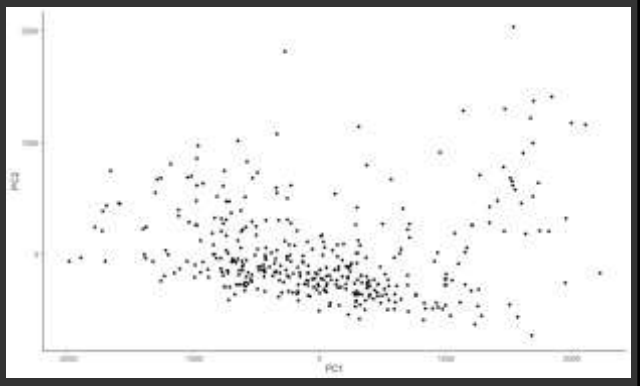
25

Again, one of the key uses of PCA is to reduce the dimensionality of your input data by capturing the information content, or variability, in a smaller number of features.

The Scree Plot provides a means to visualize the explained variance. Again, larger Eigenvalues indicate more explained variance. In the provided example, PC1 captures 81.7% of the variance, PC2 captures 15.4%, and PC3 captures 2.3%. Cumulatively, PC1 through PC3 capture 99.7% of the variance. So, six variables are reduced to three with only a slight loss in information content.

Principal Component Analysis (PCA)





	Var1	Var2	Var3	Var4	Var5	Var6
Var1	1.0000000	0.9266122	0.9199133	0.6217669	0.7892767	0.7819981
Var2	0.9266122	1.0000000	0.8871722	0.8069255	0.7784212	0.7330903
Var3	0.9199133	0.8871722	1.0000000	0.5707753	0.9069602	0.8962260
Var4	0.6217669	0.8069255	0.5707753	1.0000000	0.5598671	0.4448386
Var5	0.7892767	0.7784212	0.9069602	0.5598671	1.0000000	0.9748336
Var6	0.7819981	0.7330903	0.8962260	0.4448386	0.9748336	1.0000000

	PC1	PC2	PC3	PC4	PC5	PC6
PC1	1.000000e+00	4.486604e-16	3.936256e-16	-2.255816e-16	4.090028e-15	2.452804e-15
PC2	4.486604e-16	1.000000e+00	4.332943e-16	4.056150e-15	3.865131e-15	2.201871e-15
PC3	3.936256e-16	4.332943e-16	1.000000e+00	-1.702617e-15	-2.813617e-16	4.994810e-16
PC4	-2.255816e-16	4.056150e-15	-1.702617e-15	1.000000e+00	-2.529040e-15	4.064909e-16
PC5	4.090028e-15	3.865131e-15	-2.813617e-16	-2.529040e-15	1.000000e+00	-3.099247e-16
PC6	2.452804e-15	2.201871e-15	4.994810e-16	4.064909e-16	-3.099247e-16	1.000000e+00

26

The two graphs on this page demonstrate the concept of decorrelation. The graph on the left shows the relationship between Variable 1 and Variable 2, which have a clear correlation. In the correlation matrix, the Pearson Correlation Coefficient for these two variables 0.923. So, there is significant correlation.

In the graph on the right, I am graphing the first and second principal component. The correlation between these variables is effectively zero, ignoring some rounding error.

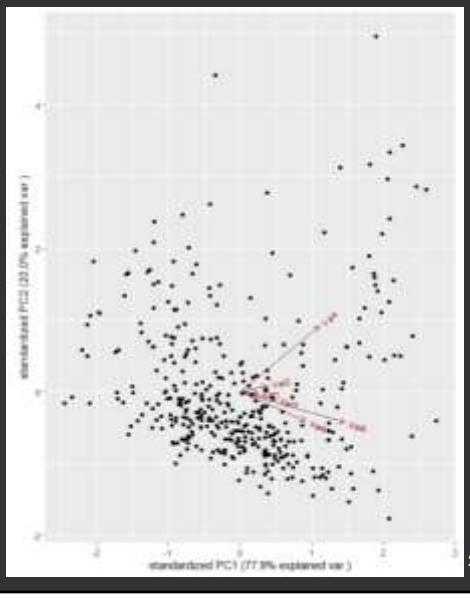
So, we are capturing variability while removing or minimizing redundant information.

Principal Component Analysis (PCA)

```
Importance of components:
                          PC1      PC2      PC3      PC4      PC5      PC6
Standard deviation     809.5542 410.7705 112.00948 56.15168 34.79928 25.56569
Proportion of Variance   0.7787   0.2005   0.01491  0.00375  0.00144  0.00078
Cumulative Proportion    0.7787   0.9791   0.99404  0.99778  0.99922  1.00000
```

```
Standard deviations (1, .., p=6):
[1] 809.55420 410.77054 112.00948  56.15168  34.79928  25.56569

Rotation (n x k) = (6 x 6):
             PC1          PC2          PC3          PC4          PC5          PC6
Var1 0.1146256 -0.007213278 -0.466466721  0.05201695 -0.4481221  0.752129368
Var2 0.1781144  0.079202334 -0.526378320  0.04057881 -0.5034498 -0.655606449
Var3 0.2327137 -0.095578242 -0.616236444 -0.31273043  0.6775808  0.006764803
Var4 0.5108610  0.836558322  0.129088331  0.09049894  0.1010786  0.064191154
Var5 0.6790373 -0.388023328  0.330032766 -0.48445177 -0.2114380  0.005005776
Var6 0.4229775 -0.366261969  0.001372416  0.80929827  0.1780066 -0.017037424
```



27

This graph shows a plot of the first two principal components. The red lines indicate the original variables. The direction of the red line is associated with the correlation between the variable and the two principal components, whereas the length of the line relates to the magnitude of correlation.

For example, based on the Eigenvectors, all six variables are positively correlated with the first principal component. That is why all the red lines are pointing to the right. In contrast, some of the original variables are positively correlated with the second principal component while others are negatively correlated. This is why some of the lines point downward and some point upward.

In the graph, each point represents one record or data point from the original dataset mapped to the principal component values.

Kernel PCA

- Use a kernel function to project the data into a higher dimensional space
- Perform PCA in this higher dimensional space

28

Kernel principal component analysis (kernel PCA) expands upon PCA by first projecting the input data into a higher dimensional space using a kernel function, such as a polynomial or radial basis function (RBF). The principal components are then calculated in this higher dimensional space where captured linear relationships can map to nonlinear relationships in the original, lower dimensional space defined by the original predictor variables.

Independent Component Analysis (ICA)

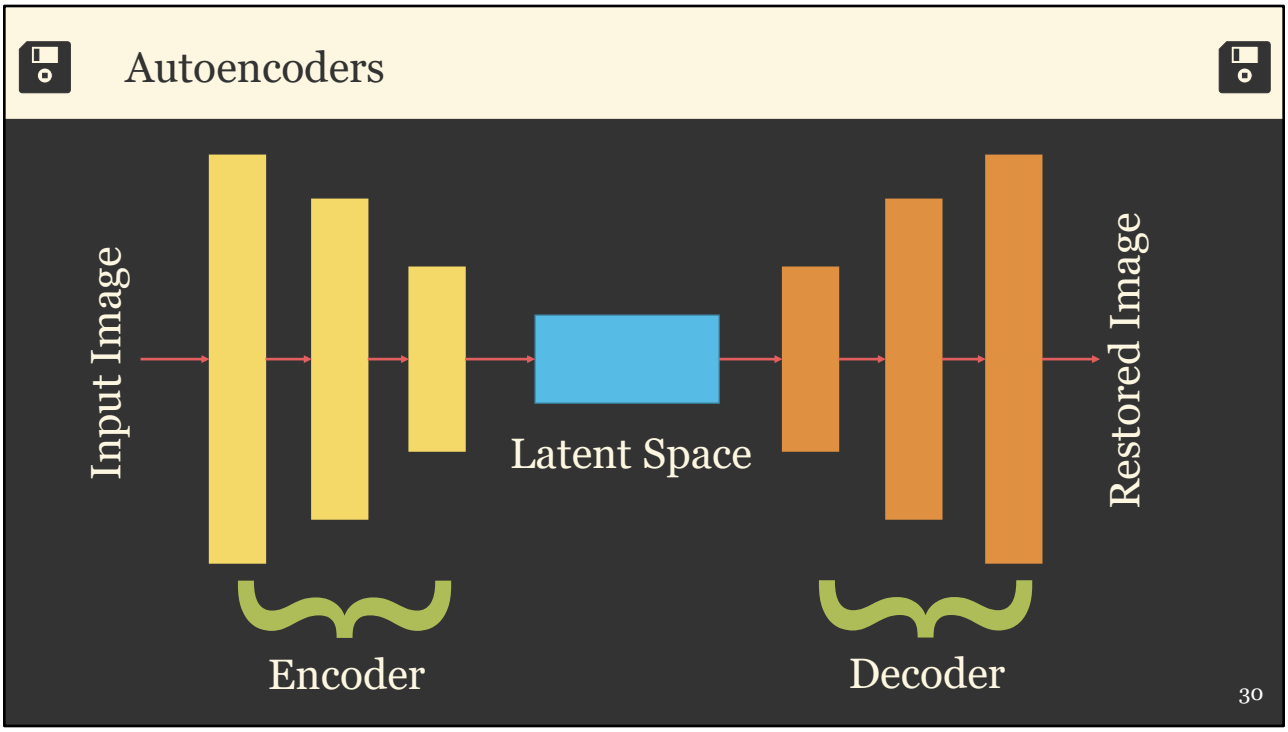
- Generate additive and statistically independent variables

29

One issue with PCA is that the resulting PC variables may still have a statistical correlation even if they are not linearly correlated.

Independent Component Analysis (ICA) generates additive and statistically independent variables from the original variables.

The ICA results are obtained by optimizing relative to a measure of statistical independence between the generated variables.

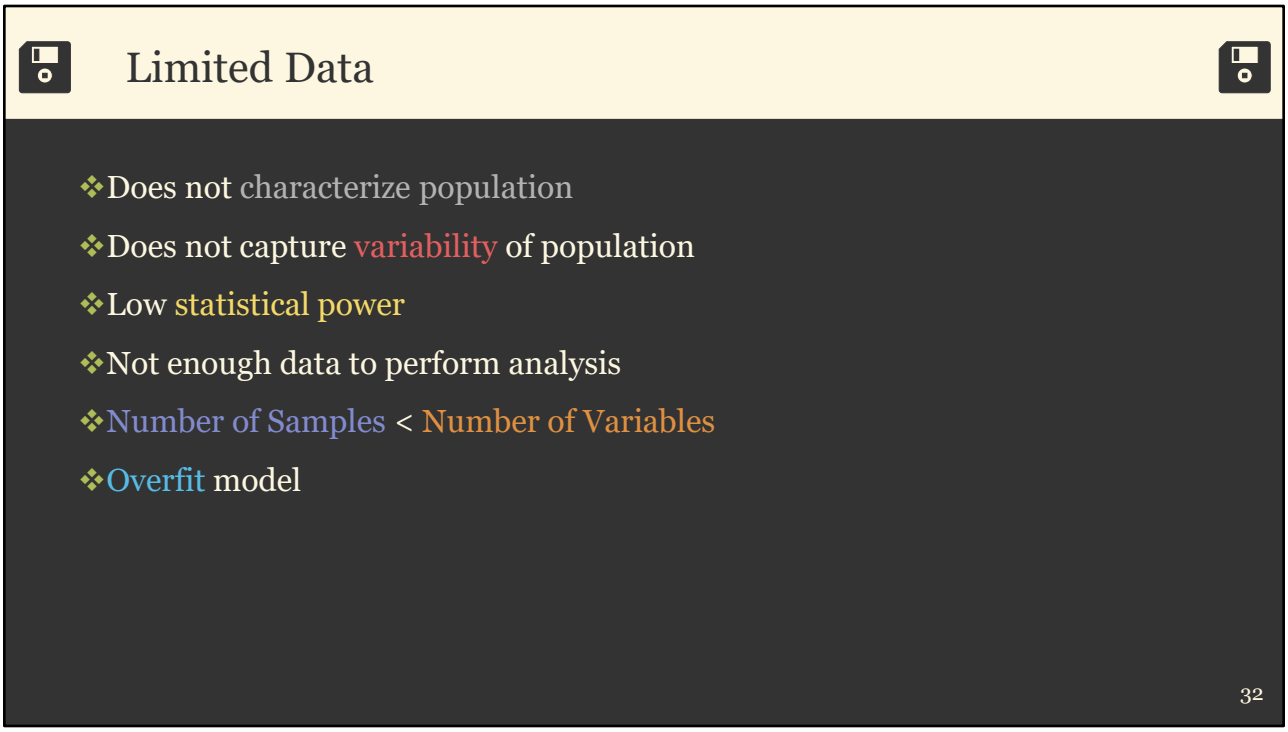


Autoencoders (AEs) are a more advanced feature reduction method that make use of artificial neural network architectures.

The original variables are the inputs to the encoder component of the architecture while the outputs of the decoder are reconstructions of the original variables. The goal of the encoder component of the architecture is to reduce the original set of input features to a lower dimensional space, which is termed the latent space. The decoder accepts the latent space representations and attempts to reconstruct the original predictor variables. The architecture's trainable parameters are updated in an attempt to minimize the difference between the predicted and original predictor variable values.

Once the model has been trained, only the encoder component is required to generate the reduced feature space. Specifically, the feature reduction is accomplished by passing the original data through the encoder to return the latent space representation. This latent space representation is the reduced feature space that is provided as input to a predictive model.

Data Volume



Another major issue is when the dataset is small or not enough samples exist to perform the required analysis. A small sample may not adequately characterize the entire population in terms of central tendency and/or variability. This can limit the power of statistical tests, a concept which will be discussed in the statistical inference section.

You may not have enough samples to perform a specific analysis. For example, in order to perform linear regression, you need to have more data points than you have predictor variables. If the number of predictor variables is larger than the number of samples, the mathematics required to determine regression coefficients cannot be performed.

In the world of machine learning, smaller training sets tend to result in less accurate models or models that are overfit to the training samples and do not generalize well to new samples. These concepts will be discussed in later modules.

In short, you may find that the data do not support the desired analysis. So, you may need to perform a different analysis that the data can support or collect more data.

Big Data

- Limited computational resources
- Storage
- Can't test, visualize, or interact with all data points
- Increased complexity/specialized skillsets
- High-performance computing
- Too much statistical power?

https://research.wvu.edu/researchers/computational-research/hpc

33

Large data sets can also be problematic. For example, very large datasets may not be able to be analyzed using a single laptop or desktop computer. You may need to have access to a computer cluster to perform your experiments or analyses. Using this type of equipment requires access to the resources and specialized knowledge.

Another issue is what statistical analyses performed on very large samples may generate small p-values due to a large degree of statistical power that may not actually be meaningful. Again, we will discuss this issue in more detail in the statistical inference module.

Summary of Key Points

- Real data can be and often is messy
- Various methods are available to prepare or engineer variables for analysis input
- Data cleaning and prep are a large component of work undertaken by data scientists
- Common pre-processing tasks:
  - Centering and scaling
  - Variable transformation to combat issues of normality
  - Detecting and dealing with outliers
  - Flagging and dealing with missing data
  - Recoding the levels of a nominal variable
  - Creating new variables from multiple input variables
- Principal Component Analysis
  - One way to create new variables from input variables
  - Results will be decorrelated, linear combinations of the input variables
  - Based on the assumption that variability represent information

This is the end of this lecture module.

Please return to the West Virginia View Webpage for additional content.

Thanks! Hope you found this useful.