Math 395/372
Project 01

1 Introduction

In this project you will learn about different methods for exploring relationships between variables in a data set. This will include applying some of the methods we have discussed in class this semester. The techniques discussed here can be applied to a variety of data sets, but we will focus on analyzing aerial imagery in this project. You will prepare a report summarizing your analysis in the form of a scientific paper. You must work in a group of 2-3 students.

2 Relationships between variables in a data set

2.1 Covariance and correlation

The *covariance* between two random variables X and Y is

$$\mathrm{Cov}(X,Y) = E\left[(X-\mu_X)(Y-\mu_Y)\right],$$

where $\mu_X = E(X)$ and $\mu_Y = E(Y)$ are the expected values (means) of the two variables. The covariance between a random variable and itself is the variance of the random variable

$$\mathrm{Var}(X) = \mathrm{Cov}(X,X).$$

The standard deviation of the random variable X is the square root of its variance

$$\sigma_X = \sqrt{\mathrm{Var}(X)}.$$

The *correlation* between two random variables X and Y is

$$\rho = \mathrm{corr}(X,Y) = \frac{\mathrm{Cov}(X,Y)}{\sigma_X \sigma_Y} = E\left[\frac{(X-\mu_X)}{\sigma_X}\frac{(Y-\mu_Y)}{\sigma_Y}\right],$$

where $\sigma_X^2 = \mathrm{Var}(X)$ and $\sigma_Y^2 = \mathrm{Var}(Y)$. Covariance and correlation both measure the strength of the linear relationship between X and Y, but correlation is normalized and dimensionless.

Given m observations $x_1, x_2, \ldots, x_m$ and $y_1, y_2, \ldots, y_m$ of the random variables X and Y, we can estimate the expected value of X using the sample mean

$$\bar{x} = \frac{1}{m}\sum_{i=1}^{m} x_i,$$

and we can estimate the covariance between X and Y using the sample covariance

$$s_{xy} = \frac{1}{m-1}\sum_{i=1}^{m}(x_i-\bar{x})(y_i-\bar{y}).$$

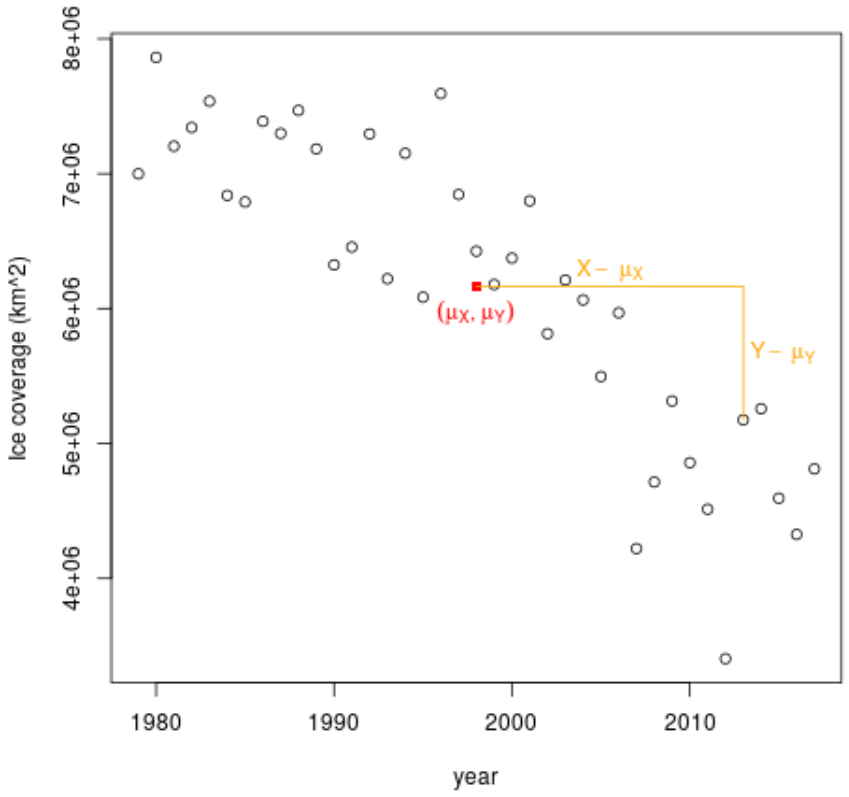

Figure 1: Ice coverage at the Arctic sea ice minimum. The mean year and ice coverage are given by μ_X and μ_Y.

To help understand why covariance measures the strength of the linear relationship between X and Y we will consider a data set that shows the decline of Arctic sea ice over time. For this data set X is the year, and Y is the total area (km^2) covered by sea ice when it is at its minimum extent for the year. The pair (X_i, Y_i) gives the ice coverage Y_i in year X_i. A scatterplot of the data is show in Figure 1. Looking at the plot, it is clear that Y tends to be lower (higher) than μ_Y when X is higher (lower) than μ_X. Thus, we expect the product $(X - \mu_X)(Y - \mu_Y)$ to tend to be negative. Thus the covariance is negative. The stronger this relationship is, the larger in magnitude the covariance will be. On the other hand, if Y tends to be higher (lower) than μ_Y when X is higher (lower) than μ_X, then $\text{Cov}(X, Y) > 0$. If X and Y are independent, then positive and negative products will tend to even out, and $\text{Cov}(X, Y) = 0$.

A similar pattern holds for the correlation. However, due to the normalization

$$-1 \leq \rho \leq 1.$$

Values of $|\rho|$ close to 1 indicates a strong linear relationship.

Let $Z^{(1)}, Z^{(2)}, \ldots, Z^{(n)}$ be a collection of n random variables. Then we can define the random vector, $\mathbf{Z} = (Z^{(1)}, Z^{(2)}, \ldots, Z^{(n)})$. If $\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_m$ are m observations of the random vector, then the *vector sample mean* is

$$\bar{\mathbf{z}} = \frac{1}{m} \sum_{i=1}^{m} \mathbf{z}_i.$$

Note that the jth element of the sample mean vector is the sample mean of the jth random

	mpg	disp	hp	wt
mpg	1.00	-0.85	-0.78	-0.87
disp	-0.85	1.00	0.79	0.89
hp	-0.78	0.79	1.00	0.66
wt	-0.87	0.89	0.66	1.00

Table 1: Correlation matrix for four *mtcars* variables.

variable, $\bar{z}^{(j)}$. The *sample covariance matrix* is an $n \times n$ matrix given by

$$\mathbf{s} = \frac{1}{m-1} \sum_{i=1}^{m} (\mathbf{z}_i - \bar{\mathbf{z}})(\mathbf{z}_i - \bar{\mathbf{z}})^T.$$

Note that the element $\mathbf{s}_{ij}$ is the sample covariance between $Z^{(i)}$ and $Z^{(j)}$.

We can simplify these formulas if we arrange the m observations of the n dimensional random vector into a *data matrix*, $\mathcal{Z} \in M_{m,n}(\mathbb{R})$, defined by

$$\mathcal{Z} = \begin{bmatrix} \mathbf{z}_1^T \\ \mathbf{z}_2^T \\ \vdots \\ \mathbf{z}_m^T \end{bmatrix}.$$

In this case, the sample mean is

$$\bar{\mathbf{z}} = \frac{1}{m} \mathcal{Z}^T \mathbf{1}_m,$$

where $\mathbf{1}_m \in M_{m,1}(\mathbb{R})$ is a vector of ones. We define $\mathbf{G}$ to be the data matrix after the column means have been subtracted out,

$$\mathbf{G} = \mathcal{Z} - \mathbf{1}_m \bar{\mathbf{z}}^T.$$

$\mathbf{G}$ is called a *centered data matrix*. Then the sample covariance matrix is

$$\mathbf{s} = \frac{1}{m-1} \mathbf{G}^T \mathbf{G},$$

and the *sample correlation matrix* is

$$\mathbf{r} = \mathbf{d}^{-1/2} \mathbf{s} \mathbf{d}^{-1/2},$$

where $\mathbf{d} \in M_n(\mathbb{R})$ is the diagonal matrix with diagonal entries $\mathbf{d}_{jj} = \mathbf{s}_{jj}$. Note that $\mathbf{s}$ and $\mathbf{r}$ are both symmetric positive definite. Recall that this implies, for example, that they have positive eigenvalues and orthogonal eigenvectors.

Figure 2 shows scatter plots for pairs of four of the variables in the *mtcars* dataset. The variables are mpg (gas mileage), disp (engine displacement), hp (horespower), wt (vehicle weight), each measured for 32 vehicles. We can view the data as 32 observations of a four dimensional random vector, $\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_{32}$. Thus, the data matrix is a 32×4 matrix, with each row representing one observation (one car), and each column representing one of the variables (the first column contains all of the gas mileage measurements). The sample correlation matrix is given it Table 2.1. Compare this with the scatter plots in Figure 2.

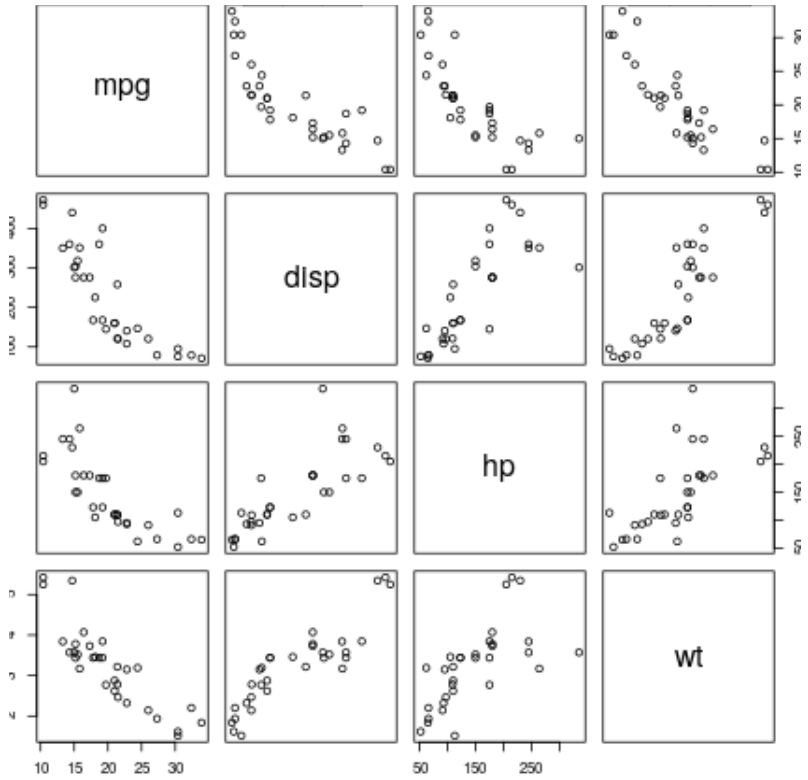

Figure 2: Scatter plots for pairs of variables in the *mtcars* dataset.

2.2 Principal components analysis

Let $\mathbf{G}$ be a centered data matrix. The observations are represented in terms of a standard basis that reflects the measurements that were taken. For example, if $\mathbf{g}_i^T = (1.2, -3.0, 4.7, -9.1)$ is the first row of $\mathbf{G}$ (the first observation), then for the first observation the first variable was 1.2 units above its mean, the second variable was 3 units below its mean, and so on. This basis is convenient, because it relates directly to the measurements. However, a better way to construct a basis for the data set is to do so in a way that best highlights the structure of the variation in the data. For example, in the *mtcars* data shown in Figure 2, the variables *mpg*, *disp*, *hp*, and *wt* are all correlated with each other and thus can be viewed as highly redundant. For example, as weight increases, horsepower and displacement tends to increase and gas mileage decreases. Is there some underlying single variable that describes most of this variation? We might imagine that large gas-guzzling cars and small economy cars are at opposite extremes when plotted on an axis representing this variable. This first principal direction is the best single direction for separating the cars from each other. When that component is subtracted out, what is the remaining direction of greatest variation? As we continue to identify and subtract out successive principal components, eventually we expect the small remaining variation to simply be due to noise.

The first principal direction ($\mathbf{v} \in \mathbb{R}^n$ such that $\|\mathbf{v}\|_2 = 1$) is the one in which the variance of

$$\mathbf{y} = \mathbf{G}\mathbf{v}$$

is maximized. Note that $\mathbf{G}\mathbf{v}$ is a linear combination of the columns of $\mathbf{G}$, and recall that each column of $\mathbf{G}$ gives the observed values of a single variable. Thus, we can think of $\mathbf{v}$ as giving a linear combination of variables that maximizes the variability in the data. The

sample variance of $\mathbf{y}$ is given by

$$\text{Var}(\mathbf{y}) = \frac{1}{m-1}\mathbf{y}^T\mathbf{y} = \frac{1}{m-1}\mathbf{v}^T\mathbf{G}^T\mathbf{G}\mathbf{v} = \mathbf{v}^T\mathbf{s}\mathbf{v},$$

where $\mathbf{s}$ is the sample covariance matrix for $\mathbf{G}$. Thus, we seek $\mathbf{v}$ that maximizes $\mathbf{v}^T\mathbf{s}\mathbf{v}$ subject to the constraint $\mathbf{v}^T\mathbf{v} = 1$.

We may use the method of Lagrange Multipliers to solve this constrained optimization problem. The solution will satisfy the system

$$\nabla(\mathbf{v}^T\mathbf{s}\mathbf{v}) = \lambda\nabla(\mathbf{v}^T\mathbf{v}), \qquad \mathbf{v}^T\mathbf{v} = 1,$$

where λ is a Lagrange multiplier. In a future homework assignment you will prove that $\nabla(\mathbf{v}^T\mathbf{s}\mathbf{v}) = \mathbf{s}\mathbf{v} + \mathbf{s}^T\mathbf{v}$. Since, $\mathbf{s}$ is symmetric, this simplifies as $\nabla(\mathbf{v}^T\mathbf{s}\mathbf{v}) = 2\mathbf{s}\mathbf{v}$. Similarly, $\nabla(\mathbf{v}^T\mathbf{v}) = 2\mathbf{v}$. Thus, the optimization problem can be reduced to

$$\mathbf{s}\mathbf{v} = \lambda\mathbf{v}, \qquad \mathbf{v}^T\mathbf{v} = 1,$$

which is simply an eigenvalue problem. Thus, the maximum must occur at a normalized eigenvector of the covariance matrix $\mathbf{s}$. At such an eigenvector, the variance is

$$\text{Var}(\mathbf{y}) = \mathbf{v}^T\mathbf{s}\mathbf{v} = \lambda\mathbf{v}^T\mathbf{v} = \lambda.$$

Consequently, the first principal direction is the eigenvector associated with the largest eigenvalue of $\mathbf{s}$.

When the first principal component is subtracted out, the remaining direction of greatest variation (the second principal direction) is the eigenvector of $\mathbf{s}$ associated with the second largest eigenvalue. The third principal direction is the eigenvector associated with the third largest eigenvalue, and so on. Since $\mathbf{s}$ is symmetric positive definite, the eigenvectors are orthonormal, and thus form a basis for $\mathbb{R}^n$.

Define $\mathbf{V}$ to be the matrix with the orthonormal eigenvectors $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n$ of $\mathbf{s}$ as columns, ordered so that $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$. The *principal components* are the columns of the matrix

$$\mathbf{Y} = \mathbf{G}\mathbf{V}.$$

The ith entry of jth principal component gives the component of the ith observation in the jth principal direction, given by the dot product

$$\mathbf{Y}_{ij} = \mathbf{g}_i\mathbf{v}_j,$$

where $\mathbf{g}_i$ is the ith row of $\mathbf{G}$ (the ith observation) and $\mathbf{v}_j$ is the jth column of $\mathbf{V}$ (the jth principal direction.

3 Multispectral imagery of the surface of the Earth

Both satellites and aircraft are used to collect imagery of the surface of the Earth. A typical sensor measures the electromagnetic radiation from the earth's surface that enters through the sensors aperture. Different sensors are sensitive to radiation of different wavelengths. The satellite imagery that you view in Google Earth, for example, often comes from satellites with a variety of spectral bands. These usually include sensors that detect blue light, green

light, red light, and near infrared radiation, among other wavelengths. The picture that you see on Google Earth combines multiple visible bands (red, green, blue is sufficient) to create a visual representation of a portion of the surface of the earth.

Multispectral imagery can be thought of as a three-dimensional array or as a collection of m two-dimensional arrays (matrices), where m is the number of bands. In this context each of these matrices is referred to as a *raster*. The ijth element of each raster corresponds to a specific region on the earths surface (usually approximately rectangular), and is often called a *cell* or a *pixel*. The value of the ijth element for a particular band gives the radiance or the reflectance of that region of the surface in that spectral band. The *reflectance* is a normalized measurement that takes into account both the radiation detected by the sensor and the incoming radiation at the surface. It gives the proportion of light that the surface reflects in that band (reflectance = outgoing radiation / incoming radiation) and is a number between 0 and 1. Unlike a photograph that you would take with your camera, the pixels are spatially referenced so that you can find their location on the surface of the earth.

There are many applications that use multispectral imagery of the surface of the earth. Vegetation absorbs red light, but reflects near infrared light. Thus, the difference between infrared and red reflectance can be used to map vegetation, and even to assess vegetation density or health. Imagery can also be used to detect changes on the surface of the earth, including the clearing of forests or the occurrence of landslides. Imagery can also be used to automatically identify/classify different landscape features.

We can think of each of the the different imagery bands as a giving a measurement of a separate variable, and we can think of each pixel as a separate observation. If m is the total number of pixels in each band and n is the number of bands, then the imagery data can be organized into an $m \times n$ data matrix $\mathcal{Z}$. Each column of $\mathcal{Z}$ represents a single band, and each row represents a single pixel, so that the ijth element of the matrix is the radiance or reflectance measured in the jth band at the ith pixel.

Imagery bands are often highly correlated. For example, areas that are illuminated tend to have higher values in all of the bands compared to areas that are in shadows. Also, healthy vegetation usually appears green during the growing season while also reflecting a lot of near infrared light. When working with imagery it is important to understand these relationships. Often it is desirable to perform a principal components analysis (PCA) of the imagery to identify principal directions and decompose the imagery into principal components. This can be especially useful for image classification, since principal components maximize variation in the imagery data and can often be used to separate different landscape features.

In this project we will analyze a multispectral image (Figure 3) of the shoreline of Lake Michigan near Holland, MI. The imagery was collected as part of the National Agriculture Imagery Program (NAIP), which is administered by the Farm Service Agency of the United States Department of Agriculture. NAIP imagery, which is collected using sensors mounted to manned aircraft, has 4 multispectral bands (in order: red, green, blue, near infrared). Recent imagery has a ground sampling distance of 0.6 m (each pixel represents a 0.6 m square on the surface of the earth). The data are reflectances in the four bands, stored in an 8-bit unsigned integer format. Zero reflectance corresponds with the value 1, and 100% reflectance corresponds with the value 255. The value 0 is reserved to represent 'no data'.

The particular image we will consider was collected during the growing season in July 2018. It includes a variety of landscape features, including open sand dunes, beach, forested dunes, and Lake Michigan. We will focus on a smaller region of interest (ROI) that is outlined in Figure 3. The ROI includes the Hope College Nature Preserve (NE corner) and

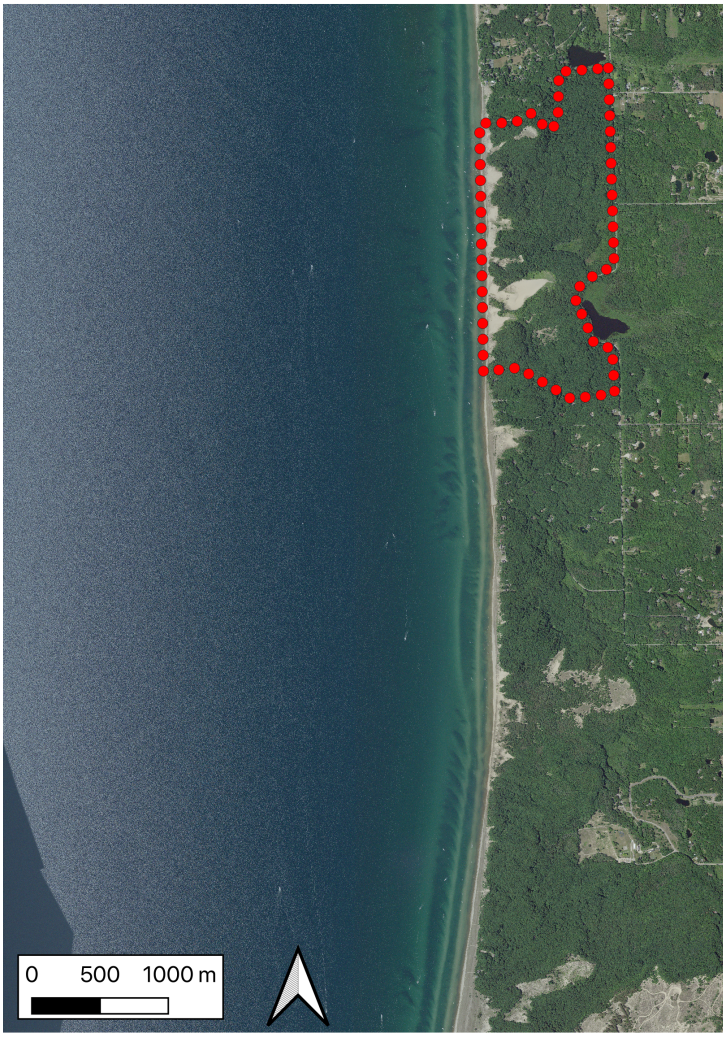

Figure 3: NAIP image (RGB) from July 2018. Region of interest outlined in red.

Green Mountain Beach dune (large active sand dune near the SW corner.

4 What you need to do

You will conduct an initial analysis of the 2018 NAIP imagery of the Lake Michigan shoreline near Holland. This will include determining and exploring the correlation between the four imagery bands, performing a PCA on the bands, and displaying a false-colored image with PC1, PC2, PC3 displayed as R, G, B, repecively. Here PC1 refers to the first principal component, etc. You will also explore how well the first two principal components separate different landscape features in the imagery.

You will write a report that describes the analysis. The report should have the structure of a scientific paper with the following sections:

- Introduction. A brief introduction describing the problem.
- Methods. Describe the data and the techniques used in the analysis. Do not include results of the analysis in the methods section. This should include describing the

programs that you write, but should not include the actual code. When you describe the PCA, summarize its relationship to the SVD, referring to the appendix for details.

- Results. Describe the results of the analysis, presenting any figures and tables of results. Each figure or table should have a caption, and each one should be referred to in the text.

- Discussion. A brief discussion section discussing the results and their potential usefulness to someone who might want to use imagery to study this region.

- Appendix. You should include an appendix that addresses the following: PCA is closely related to the SVD. Determine the relationship between the eigenvalues of the covariance matrix $\mathbf{s}$ and the singular values of $\mathbf{G}$. Also determine the relationship between the eigenvectors of the covariance matrix $\mathbf{s}$ and the right singular vectors of $\mathbf{G}$. Justify these relationships. Hint: Start by replacing $\mathbf{G}$ with its reduced SVD, in the product $\mathbf{G}^T\mathbf{G}$ and simplifying.

Type all of the text in the report, including any mathematics. Mathematics should be nicely formatted. If you know how to use a Latex editor, I encourage you to do so. Otherwise, you should use MS Word. Part of your grade will depend on the quality of your writing and your adherance to the specified report structure.

You should only make one submission per group. You should submit two files from one group member's Moodle account. The first should be your report in either MS Word or PDF format. The second should be a zip file containing auxillary files. These must include a jupyter notebook that contains all of your analysis, any python scripts that your jupyter notebook uses, and the .csv file you create. I should be able to place your notebook, scripts, and .csv file into a directory with the scripts and other files I have provided you and repeat your analysis by just running the code in the notebook. The jupyter notebook should include text blocks briefly describing each code chunk.

Here are some specific requirements for your report/analysis:

- In the methods section you should include a figure that shows the ROI. The imagery should be cropped and masked to only show this region. I have given you a python script that produces a cropped and masked GeoTiff file and also plots the image using matplotlib. You may use either of these to create the figure. If you have experience with GIS, you may want to use the GeoTiff and create the figure in something like QGIS. If you do this, make sure that the fourth band is not being interpreted as an alpha channel.

- The raster input and output are handled by a python script that I have provided for you. You will need to install the rasterio and geopandas packages in your numa environment in order to run the scripts. The script returns the data matrix for the ROI in the format described in Section 3, *but it is not centered*. Before you perform the PCA or find the covariance matrix or correlation matrix *you need to center it*.

- Using the data matrix (before centering it), produce a plot similar in structure to Figure 2, showing scatter plots for each pair of bands. The plot should be made up of 16 subplots in an array. You should use the subplots function in pyplot. In addition to the scatter plots, plot the corresponding best-fit line (in the least squares sense) on the same axes. Make sure you can clearly see the line along with the scatter plot in each

subplot. Use your own QR factorization routine to solve the least squares problems (you should use loops to do this for the 16 problems, and you should use loops to create the 16 subplots). Your QR factorization should use Householder reflections. Your scatter plots will be too crowded if you include all of the points, so just plot a subset of 50 randomly selected points in each scatter plot. However, your linear models should be fit using the entire data set–not just 50 points.

- Create two tables giving the results of the linear fits. One table should give the slopes, and one table should give the intercepts. The table should have the structure of table 2.1, with the ij entry giving the slope or the intercept of the regression line relating variable i and variable j. However, unlike table 2.1, these tables should not be symmetric.

- After creating the centered data matrix $\mathbf{G}$, compute the covariance matrix and the correlation matrix, and present them as separate tables. The tables should have the structure of table 2.1.

- Using $\mathbf{G}$, perform a PCA. Use the eigh function in numpy to find the eigenvalues. The eigh function works with real symmetric or complex Hermitian matrices. Create an $m \times n$ matrix containing the pricipal components as columns (make sure they are in the correct order). $n = 4$ is the number of bands, and $m = 5185371$ is the number of pixels in the ROI. Compute the correlation matrix for the principal components and report it in a table. Is it what you expect it to be?

- I have provided you with a script that will display a false color image by treating the first three principal components as red, green, and blue values. The script also creates and saves a GeoTiff file of the results. Use the script to create the false color image and create a figure that displays the true color image and the false color image side by side. In your results section you should comment on the differences that the principal components image seems to highlight, and in your discussion section comment on how that information might be used in practice.

- If you run the command %matplotlib notebook in your jupyter notebook before you run the sripts to produce the imagery plots, the pyplot plots will be interactive. You can zoom and pan the plots, and as you move the pointer over a plot, the coordinates and band values will be displayed. We will take advantage of this feature to manually extract coordinates and band values for pixels representing different landscape classes. For the original RGB bands, identify 10 pixels in each of the following classes: bare sand, grass-covered sand dune, and illuminated forest canopy. At each pixel, record the coordinates, the three RGB band values, and the class. Using the coordinates, identify the same points in the principal components image and record the values of the first three principal components (PC1,PC2,PC3). You should create a spreadsheet with a row for each pixel and 10 columns. The first column should give the point number (1-30), the second and third columns should give the point location, the fourth through sixth columns should give the RGB values, the seventh through ninth columns should give the PC1-3 values, and the last column should give the class. You may work with other groups to complete the spreadsheet. Save your spreadsheet as a .csv file.

- Import the spreadsheet you created and use it to create two scatter plots using pyplot. The first should show the 30 points in the R,G plane. The second should show the 30

points in the PC1,PC2 plane. Use distinct symbols and colors as markers for the three different classes. In your results section evaluate whether R,G or PC1,PC2 are more useful for separating the classes from each other, and comment on the utility of this result in your discussion section. You may need to zoom in on different portions of the plot to adequately address this. **I have not done this part of the project, and I am very interested to see your results!**