



Perform PCA for Dimensions 6 to 10 of Chile's Tourism Regions

Yue Sun
Reshma Patil



Roadmap



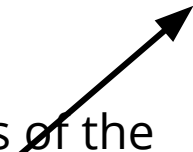
Data Cleaning

PCA

Recommendation

Data Cleaning

1. Delete first 6 empty rows of the Excel data file
2. Save the data file as csv file
3. Read the csv file by Python



1				
2				
3				
4				
5				
6				
8	Regions	Density of restaurants per 100,000 habitants	People working at restaurants per 10,000 habitants	Car r
9	Arica y Parinacota (XV)	35.857	19.563	
10	Tarapacá (I)	25.947	26.616	
11	Antofagasta (II)	37.248	35.446	
12	Atacama (III)	44.823	26.933	
13	Coquimbo (IV)	37.632	25.165	
14	Valparaíso (V)	52.408	47.050	
15	Metropolitana (RM)	9.223	16.914	
16	O'Higgins (VI)	18.959	49.422	
17	Maule (VII)	15.637	19.943	
18	Biobío (VIII)	14.128	14.004	
19	Araucanía (IX)	15.986	15.687	
20	Los Ríos (XIV)	34.512	26.207	
21	Los Lagos (X)	36.415	27.360	
22	Aysén (XI)	76.500	40.087	

Data Cleaning

4. Remove useless rows

5. Rename the first column that represents regions and make sure the names of regions are the same as those from dim1 to dim 6 so that combining the PCA results in the end will be easy.

```
: # Remove last rows - from 16 to 18  
chile_data_2 = chile_data_2[:-4]
```

```
: # Rename the first column  
chile_data_2 = chile_data_2.rename(columns={'Regions': 'Region'})
```

```
: chile_data_2
```

	Region	Density of restaurants per 100,000 habitants	People working at restaurants per 10,000 habitants	Car rental agencies	Hospital beds per 10,000 habitants	Density of ATM machines per 100,000 habitants	Spas	Ca 10 hak
0	Arica y Parinacota	35.857	19.563	12	21.619	43.766	0	
1	Tarapacá	25.947	26.616	2	18.958	60.264	0	
2	Antofagasta	37.248	35.446	22	29.96	69.233	3	
3	Atacama	44.823	26.933	19	25.478	56.225	1	
4	Coquimbo	37.632	25.165	9	17.805	47.745	3	
5	Valparaíso	52.408	47.05	16	27.756	55.85	9	
6	Metropolitana	9.223	16.914	47	25.508	65.499	0	

Data Cleaning

6. Remove extra symbols

7. Delete columns in which 30% ~ 70% of the variables are NaNs.

8. Unlist columns

```
# Remove $ symbol
chile_data_2 = chile_data_2.replace(r'[\<$]', '', regex = True)
# Remove commas from numbers
chile_data_2 = chile_data_2.replace(',', '', regex = True)
# Remove `~` character
chile_data_2 = chile_data_2.replace('-', '', regex = True)
# Replace empty values with NaNs
chile_data_2 = chile_data_2.replace(r'^\s*$', np.nan, regex = True)
# Check NaNs in the dataset again
chile_data_2
```

```
# Impute data in four columns
imputer = SimpleImputer(missing_values = np.nan)
chile_data_2[['Crime index',
              'Illegal commerce',
              'Seed funds allocated to the tourism sector ($)',
              'Yearly budget for international tourism promotion ($)'],
             [['Crime index',
              'Illegal commerce',
              'Seed funds allocated to the tourism sector ($)',
              'Yearly budget for international tourism promotion ($)']]]
```

```
# Drop 4 columns with many NaNs
chile_data_2 = chile_data_2.drop(['Ski resorts',
                                  'Major shopping centers',
                                  'Number of vineyards',
                                  'Governmental resources a
```

```
# Setting index
chile_data_2 = chile_data_2.set_index('Region')

# Select columns
cols = chile_data_2.loc[:, chile_data_2.dtypes == np.object].columns

# Convert to numeric
chile_data_2[cols] = chile_data_2[cols].apply(pd.to_numeric, errors='coerce')

# Now all our columns are integers or floats
chile_data_2 = chile_data_2.reset_index()
```

Standardize data for applying PCA

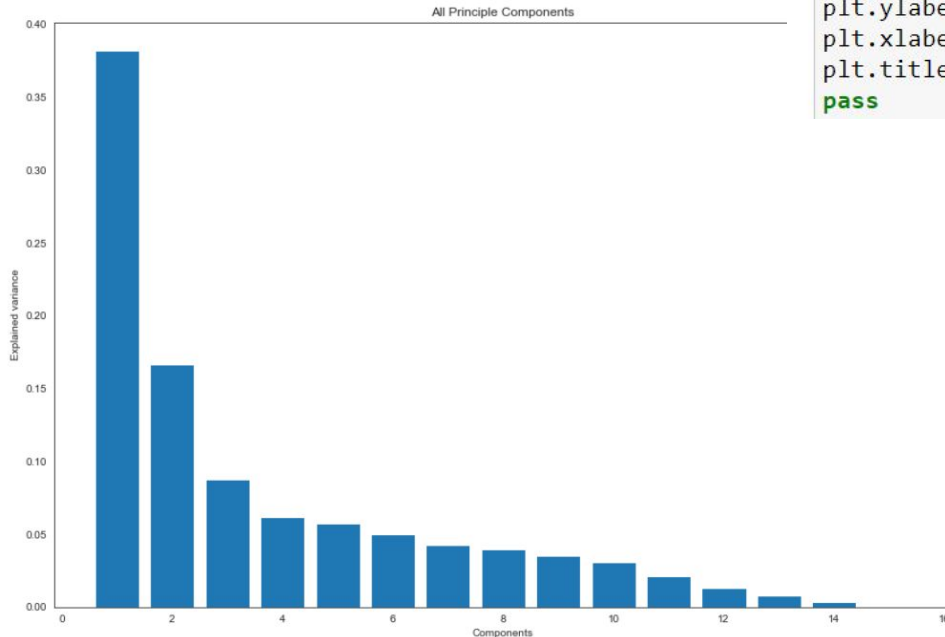
```
# Create a copy
chile_data_s_2 = chile_data_2.copy()

# Standardize
scaler = StandardScaler()
chile_data_s_2.loc[:, chile_data_s_2.columns != 'Region'] = scaler.fit_transform(chile_data_s_2.loc[:,
                                                                                     chile_data_s_2.columns != 'Region'])

# Set region as an index column
chile_data_s_2 = chile_data_s_2.set_index('Region')
pass
```

PCA

1. Eigenvalues & Eigenvectors
2. Run PCA and fit the model



```
: # Run PCA and fit the model
myPCA = PCA()
x = myPCA.fit(chile_data_s_2)

# Plotting the variance explained by each component
plt.bar(range(1, len(x.explained_variance_) + 1), x.explained_variance_ratio_)
plt.ylabel('Explained variance')
plt.xlabel('Components')
plt.title('All Principle Components')
pass
```

3. Plot the variance explained by each component

PCA

3. Explore the importance of each feature for principle components

```
pca_model = myPCA.fit_transform(chile_data_s_2)
PCcomponents = pd.DataFrame(data = pca_model, columns = ['PC1', 'PC2', 'PC3', 'PC4', 'PC5', 'PC6', 'PC7'])
print("\n The Factor scores are")
PCcomponents
```

4. Calculate factor scores

The Factor scores are

	PC1	PC2	PC3	PC4	PC5	PC6	PC7
0	-2.373870	0.823256	0.472329	-2.079397	-2.919301	0.141580	0.186033
1	-1.485061	-0.247090	6.228244	-2.115655	1.203540	0.595868	0.757801
2	0.263871	2.084163	1.119329	4.384186	0.480078	3.195863	0.422146
3	-1.814625	0.019734	2.139679	2.229493	-0.772417	-1.927273	-0.772461
4	-0.969405	-1.998904	-0.011820	0.458845	1.718456	-1.177366	-1.155089
5	1.188020	0.340730	-0.711721	-0.291174	-0.750611	1.368108	0.495793
6	14.460293	2.855751	0.166888	-0.525576	-0.548933	-0.744969	-0.624666
7	-1.283626	-0.937935	-0.789281	-1.051599	-2.088445	1.379180	-1.114501
8	-0.680583	-3.431526	-0.366983	-0.213786	-1.694869	-0.554999	-0.381959
9	2.024578	-3.385638	-1.595265	-1.373500	2.088322	0.846775	3.060953
10	0.412162	-4.107222	-0.063790	1.987826	-0.289221	-2.512417	0.329350
11	-1.888205	-1.361590	-2.279907	0.261054	-1.076617	1.509792	-0.109209
12	0.031550	-0.800397	-1.746730	-0.594429	3.044853	-0.001577	-1.242050
13	-3.898966	3.945054	-0.990212	-1.273527	1.729195	0.068358	-2.559529
14	-3.986135	6.201613	-1.570759	0.197240	-0.124031	-2.186923	2.707388

```
pca = PCA(n_components = 7).fit(chile_data_s_2)
vars = pca.explained_variance_ratio_
c_names = chile_data_s_2.columns
sum = 0

print('Variance: Projected dimension')
print('-----')
for idx, row in enumerate(pca.components_):
    output = '{0:4.1f}%: '.format(100.0 * vars[idx])
    output += " + ".join("{0:5.2f} * {1:s}".format(val, name) \
                          for val, name in zip(row, c_names))
    sum += 100*vars[idx]
    print(output)

print('Total variance explained by the 7 components {0:4.1f}%'.format(sum))
# Total variance explained by the 7 components 84.8.0%
```


PCA

5. Fit the model

6. Compute weights

```
myPCA = PCA(n_components = 7)
pca_model = myPCA.fit(chile_data_s_2)
y_axis = [0,0,0,0,0,0,0]
for i in range(0,7):
    y_axis[i]=[np.mean(pca_model.components_[i][0:15]), np.mean(pca_model.co
        np.mean(pca_model.components_[i][27:36]), np.mean(pca_model.co
        np.mean(pca_model.components_[i][41:46]))]

# Plot
x_axis = ['TOURISM-RELATED SERVICES', 'SECURITY AND SAFETY ', 'ECONOMIC PERI
plt.plot(x_axis,y_axis[0], color = 'mediumaquamarine', label = "C1")
plt.plot(x_axis,y_axis[1], color = 'yellow', label = "C2")
plt.plot(x_axis,y_axis[2], color = 'pink', label = "C3")
plt.plot(x_axis,y_axis[3], color = 'steelblue', label = "C4")
plt.plot(x_axis,y_axis[4], color = 'salmon', label = "C5")
plt.plot(x_axis,y_axis[5], color = 'red', label = "C6")
plt.plot(x_axis,y_axis[6], color = 'orange', label = "C7")
plt.xticks(rotation = 90)
plt.title('Example of variable contributions to each principal component')
plt.legend()
pass
```

```
# Creating a dataframe of weights
weights = pd.DataFrame(np.column_stack((chile_data_s_2.columns, pca_model.components_[0] *
    pca_model.explained_variance_ratio_[0],
    pca_model.components_[1] * pca_model.explained_variance_ratio_[1],
    pca_model.components_[2] * pca_model.explained_variance_ratio_[2],
    pca_model.components_[3] * pca_model.explained_variance_ratio_[3],
    pca_model.components_[4] * pca_model.explained_variance_ratio_[4],
    pca_model.components_[5] * pca_model.explained_variance_ratio_[5],
    pca_model.components_[6] * pca_model.explained_variance_ratio_[6]))).

weights = weights.set_index(0)

# Create a weighted average
weights['weighted_average'] = weights.sum(axis = 1)/np.sum(pca_model.explained_variance_ratio_)
|
# Print
weights.head()
```

	1	2	3	4	5	6	7	weighted_average
0								
Density of restaurants per 100,000 habitants	-0.0508779	0.0383434	-0.00603016	0.00173496	0.00703431	-0.000705732	-0.00638432	-0.019904
People working at restaurants per 10,000 habitants	-0.0371438	0.0379041	-0.00768655	0.00145828	-0.00263994	0.00488203	0.000651091	-0.003035
Car rental agencies	0.0748317	0.0115487	-0.0072556	0.0104844	0.00355798	-0.00633721	-0.00327852	0.098485
Hospital beds per 10,000 habitants	-0.00384581	0.0285705	-0.0193489	0.0143164	-0.00104324	0.00105819	0.0132646	0.038865
Density of ATM machines per 100,000 habitants	0.0220336	0.0438976	0.0153522	0.0128237	0.00260291	0.000892908	0.0075618	0.123961

PCA

```
# Ranking for dimension 6: TOURISM-RELATED SERVICES

# Create a dataframe for relevant variables
dim6 = chile_data_s_2.iloc[:, 0:15].mul(weights['weighted_average'][0:15], axis = 1)

# Create a score ranking
dim6['Ranking 6'] = dim6.sum(axis = 1)

# Sort by score
dim6.sort_values(by = 'Ranking 6', ascending = False)
```

7. Create a score ranking for dim 6

8. Compute ranks for dim 7 to 10 in the same way

9. Combine all 5 ranks

	Ranking 6	Ranking 7	Ranking 8	Ranking 9	Ranking 10
Region					
Arica y Parinacota	-0.219562	-0.431607	-0.218964	-0.127449	-0.193358
Tarapacá	-0.139428	0.184253	0.111514	-0.033586	-0.196716
Antofagasta	0.367888	0.126460	0.436016	0.197444	0.076666
Atacama	-0.059748	-0.323617	-0.149980	0.054979	-0.156984
Coquimbo	-0.327888	-0.296003	-0.271570	0.128133	-0.040475
Valparaíso	0.339170	0.177603	0.047514	-0.061269	0.058999
Metropolitana	1.793741	2.208516	1.618444	0.672550	0.647190
O'Higgins	-0.319237	-0.415101	-0.064561	-0.184920	-0.053559
Maule	-0.494721	-0.279228	-0.305912	-0.114380	-0.006079
Bío-Bío	-0.044464	0.453489	-0.071235	-0.087292	0.079230
Araucanía	-0.402073	0.026774	-0.323501	0.013958	0.052154
Los Ríos	-0.381948	-0.423620	-0.316565	-0.131099	-0.070488
Los Lagos	-0.127404	-0.209206	-0.038840	-0.032185	0.184970
Aysén	-0.302594	-0.322888	-0.210162	-0.088698	-0.261150
Magallanes y Antártica	0.318268	-0.475825	-0.242200	-0.206185	-0.120398

10. Combine the ranks for Dimensions 1 to 10

```
final_scoring_data = pd.concat([dim1.iloc[:, -1:], dim2.iloc[:, -1:], dim3.iloc[:, -1:], dim4.iloc[:, -1:], dim5.iloc[:, -1:],
                                dim6.iloc[:, -1:], dim7.iloc[:, -1:], dim8.iloc[:, -1:], dim9.iloc[:, -1:],
                                dim10.iloc[:, -1:]], axis = 1)
final_scoring_data
```

	Ranking 1	Ranking 2	Ranking 3	Ranking 4	Ranking 5	Ranking 6	Ranking 7	Ranking 8	Ranking 9	Ranking 10
Region										
Arica y Parinacota	-0.415952	0.084054	-0.227440	-0.142092	0.074729	-0.219562	-0.431607	-0.218964	-0.127449	-0.193358
Tarapacá	-0.173136	-0.013629	-0.063857	-0.051780	-0.111462	-0.139428	0.184253	0.111514	-0.033586	-0.196716
Antofagasta	-0.023722	-0.210598	-0.075573	0.003949	-0.066772	0.367888	0.126460	0.436016	0.197444	0.076666
Atacama	-0.315089	-0.340057	-0.188900	-0.203309	-0.415831	-0.059748	-0.323617	-0.149980	0.054979	-0.156984
Coquimbo	-0.250916	0.091873	-0.128560	0.049329	-0.217581	-0.327888	-0.296003	-0.271570	0.128133	-0.040475
Valparaíso	0.692363	0.174514	0.249814	0.405196	0.380127	0.339170	0.177603	0.047514	-0.061269	0.058999
Metropolitana	2.505190	0.512320	1.245244	1.013256	1.231514	1.793741	2.208516	1.618444	0.672550	0.647190
O'Higgins	-0.571647	-0.122728	-0.138538	-0.318736	-0.295041	-0.319237	-0.415101	-0.064561	-0.184920	-0.053559
Maule	-0.476902	-0.145830	-0.227193	-0.242213	-0.240416	-0.494721	-0.279228	-0.305912	-0.114380	-0.006079
Bío-Bío	0.312954	-0.259354	0.138733	-0.096586	0.019311	-0.044464	0.453489	-0.071235	-0.087292	0.079230
Araucanía	-0.510915	-0.330403	-0.159923	0.004307	-0.067661	-0.402073	0.026774	-0.323501	0.013958	0.052154
Los Ríos	-0.294472	0.045124	-0.245964	-0.086307	-0.333723	-0.381948	-0.423620	-0.316565	-0.131099	-0.070488
Los Lagos	0.204969	0.577425	0.254603	0.254766	0.565572	-0.127404	-0.209206	-0.038840	-0.032185	0.184970
Aysén	-0.441574	-0.012318	-0.253732	-0.422402	-0.465189	-0.302594	-0.322888	-0.210162	-0.088698	-0.261150
Magallanes y Antártica	-0.241153	-0.050395	-0.178714	-0.167379	-0.057579	0.318268	-0.475825	-0.242200	-0.206185	-0.120398

PCA

11. Highlight dataframe values with colors

```
# attach CSS classes to each cell
final_scoring_data.style.highlight_null().render().split('\n')[:10]

# Create a function for negative values (red)
def color_negative_red(val):
    """
    Takes a scalar and returns a string with
    the css property ``color: red`` for negative
    strings, black otherwise.
    """
    color = 'red' if val < 0 else 'black'
    return 'color: %s' % color

# Create a function for max values (yellow)
def highlight_max(s):
    """
    highlight the maximum in a Series yellow.
    """
    is_max = s == s.max()
    return ['background-color: yellow' if v else '' for v in is_max]

# Apply styles
final_scoring_data.style.\
    applymap(color_negative_red).\
    apply(highlight_max)
```

	Ranking 1	Ranking 2	Ranking 3	Ranking 4	Ranking 5	Ranking 6	Ranking 7	Ranking 8	Ranking 9	Ranking
Region										
Arica y Parinacota	-0.415952	0.084054	-0.227440	-0.142092	0.074729	-0.219562	-0.431607	-0.218964	-0.127449	-0.1933
Tarapacá	-0.173136	-0.013629	-0.063857	-0.051780	-0.111462	-0.139428	0.184253	0.111514	-0.033586	-0.1967
Antofagasta	-0.023722	-0.210598	-0.075573	0.003949	-0.066772	0.367888	0.126460	0.436016	0.197444	0.0766
Atacama	-0.315089	-0.340057	-0.188900	-0.203309	-0.415831	-0.059748	-0.323617	-0.149980	0.054979	-0.1569
Coquimbo	-0.250916	0.091873	-0.128560	0.049329	-0.217581	-0.327888	-0.296003	-0.271570	0.128133	-0.0404
Valparaíso	0.692363	0.174514	0.249814	0.405196	0.380127	0.339170	0.177603	0.047514	-0.061269	0.0589
Metropolitana	2.505190	0.512320	1.245244	1.013256	1.231514	1.793741	2.208516	1.618444	0.672550	0.6471
O'Higgins	-0.571647	-0.122728	-0.138538	-0.318736	-0.295041	-0.319237	-0.415101	-0.064561	-0.184920	-0.0535
Maule	-0.476902	-0.145830	-0.227193	-0.242213	-0.240416	-0.494721	-0.279228	-0.305912	-0.114380	-0.0060
Bío-Bío	0.312954	-0.259354	0.138733	-0.096586	0.019311	-0.044464	0.453489	-0.071235	-0.087292	0.0792
Araucanía	-0.510915	-0.330403	-0.159923	0.004307	-0.067661	-0.402073	0.026774	-0.323501	0.013958	0.0521
Los Ríos	-0.294472	0.045124	-0.245964	-0.086307	-0.333723	-0.381948	-0.423620	-0.316565	-0.131099	-0.0704
Los Lagos	0.204969	0.577425	0.254603	0.254766	0.565572	-0.127404	-0.209206	-0.038840	-0.032185	0.1849
Aysén	-0.441574	-0.012318	-0.253732	-0.422402	-0.465189	-0.302594	-0.322888	-0.210162	-0.088698	-0.2611
Magallanes y Antártica	-0.241153	-0.050395	-0.178714	-0.167379	-0.057579	0.318268	-0.475825	-0.242200	-0.206185	-0.1203

PCA

12. Comments on regions

main strengths and opportunity areas

Metropolitana: The region performs comprehensively well in all 10 dimensions. Its main strengths are in dim 1 CULTURAL HERITAGE AND EVENTS and in dim 7 SECURITY AND SAFETY. The reason is that ranking scores of Metropolitana in these dimensions are over 2.2 while those of other regions are only 0.1 or below 0. The opportunity area is NATURAL RESOURCES AND SUSTAINABILITY. Low scores are in % OF LAND THAT CORRESPONDS TO FORESTS, NATIONAL PROTECTED SITES (%) and PRESERVED SITES and SEASHORE PROTECTED SITES. Need more awareness and investment in these areas to help improve the sustainability of Metropolitana.

PCA

12. Comments on regions

Los Lagos: This region has a best score in the ranking of NATURAL RESOURCES AND SUSTAINABILITY, which is its strength. It also does well in TOURISM MOBILITY AND TRANSPORTATION INFRASTRUCTURE. The opportunity areas are in TOURISM-RELATED SERVICES, SECURITY AND SAFETY, ECONOMIC PERFORMANCE and TOURISM PROMOTION.

Valparaiso: This region performs above the average except in TOURISM PROMOTION, which is the opportunity area. Specific details are Number of tourism information offices and Yearly budget for international tourism promotion (\$M). Invest on these areas will be useful.

Recommendation

Previously, my team's recommendation is to invest more on natural sustainability and transportation infrastructure. This is in a broad view. My current recommendation is that get rid of some of the regions that contribute to tourism and focus more on some specific regions. Regions such as Aysén, Maule, O'Higgins perform poorly comprehensively. They might not be a suitable place for tourism. In terms of advertisement, the government should invest more on publicizing Metropolitana, Valparaíso, Los Lagos to attract tourists in the world. These regions should be the main focus and flagship tourism regions. Meanwhile, invest more on sustainability, SECURITY AND SAFETY and TOURISM PROMOTION on these areas because these areas have a relatively low scores, and improving them is achievable by more budget.

To sum up, I would say invest more on sustainability on regions that have a good natural resource, which is a base advantage, such as Los Lagos, advertise more on flagship regions Metropolitana, Valparaíso and Los Lagos, build more tourism related infrastructure and transportation infrastructure on these areas to accommodate more tourists, get rid of regions such as Aysén, Maule, O'Higgins as tourism places.