ISMET OKAN CELIK CWID:10472265

Question-1

In [1]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sn
from pandas.plotting import parallel_coordinates
import numpy as np
from pandas.plotting import table
import statsmodels.api as sm
```

In [2]:
```
data=pd.read_csv('EE627A_HW1_Data.csv')
data_prep=data.drop(['Date'],axis=1,inplace=False)
df=pd.DataFrame(data_prep)
CorrMatrix=df.corr()
CorrMatrix.head()
```

Out[2]:

	Mkt-RF	SMB	HML	RF	Mom	Food	Beer	Smoke	Games
Mkt-RF	1.000000	0.326863	0.216145	-0.068723	-0.338343	0.835924	0.707673	0.584268	0.830211
SMB	0.326863	1.000000	0.094113	-0.059640	-0.164023	0.201698	0.351039	0.103154	0.412089
HML	0.216145	0.094113	1.000000	0.012115	-0.400635	0.215132	0.214982	0.171809	0.250387
RF	-0.068723	-0.059640	0.012115	1.000000	0.039130	0.032222	-0.011277	0.063036	-0.024963
Mom	-0.338343	-0.164023	-0.400635	0.039130	1.000000	-0.280289	-0.200077	-0.219165	-0.356992

5 rows × 35 columns

In [3]:
```
#Dropping Factors Columns From Data Frame and Only Keeping Industries
#Because we only need correlation between factors and industries
factor_df=pd.DataFrame(CorrMatrix)
factor_df=factor_df.iloc[0:5,5:]
factor_df.head()
```

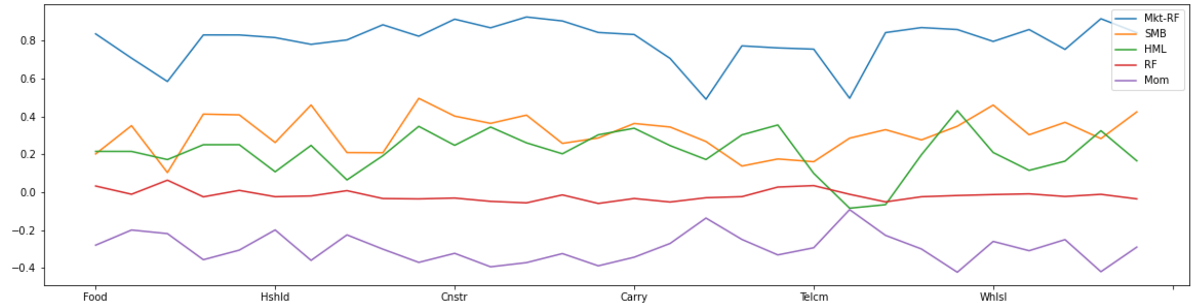
Out[3]:

	Food	Beer	Smoke	Games	Books	Hshld	Clths	Hlth	Chems
Mkt-RF	0.835924	0.707673	0.584268	0.830211	0.830092	0.816234	0.780630	0.804022	0.883889
SMB	0.201698	0.351039	0.103154	0.412089	0.408145	0.261883	0.460134	0.208896	0.208012
HML	0.215132	0.214982	0.171809	0.250387	0.250608	0.107373	0.246719	0.064393	0.192392
RF	0.032222	-0.011277	0.063036	-0.024963	0.009172	-0.024209	-0.020186	0.007783	-0.033856
Mom	-0.280289	-0.200077	-0.219165	-0.356992	-0.306399	-0.199683	-0.360482	-0.225778	-0.301203

5 rows × 30 columns

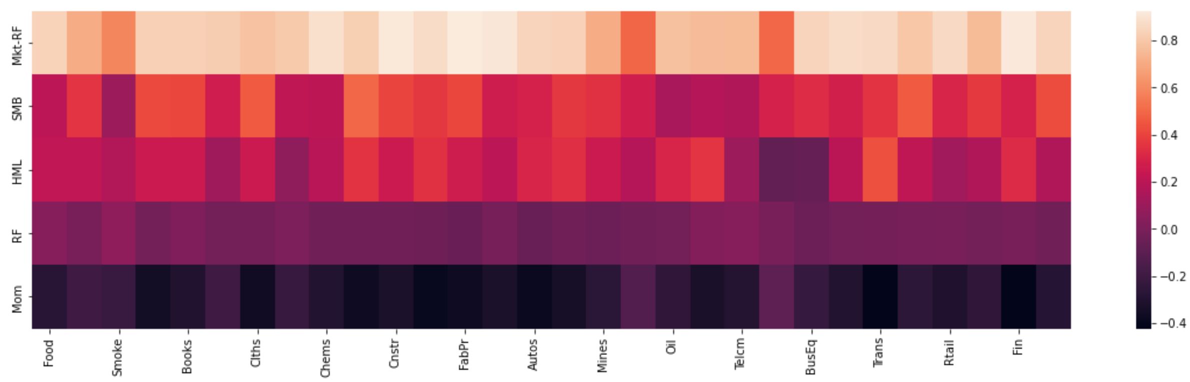
In [4]:
```
fig, ax = plt.subplots(1, 1, figsize=(20, 5))

factor_df.T.plot(table=False, ax=ax);
```



We can also see which factor most correlated with every industry on the graph. Mkt-RF (Market Risk Free) is the most correlated factor, and Mom (Momentum) is the negatively correlated factor. RF (Riskfreerate) does not correlate highly with any industry.

In [5]:
```
fig=plt.figure()
sn.heatmap(factor_df)
fig.set_figwidth(20)
fig.set_figheight(5)
plt.show()
```



Mkt_RF (Market Risk-Free) is the most highly correlated factor with every industry, as we can see on the heatmap above. The graph shows its correlation coefficient is closest to the r=1 that is why the heat color of Mkt_RF is primarily light-colored.

Mom (Momentum) is the negatively correlated factor on the heatmap. The graph shows that correlation coefficients r are around -0.4. For this reason, the heat map shows dark colors.

RF (Risk Free Rate) does not correlate with any industry, and as we can see on the graph r correlation coefficents r are around 0.

In [6]:
```
#Calculation of Auto-Correlation Function (ACF) from Time-Lag(1) to Time-Lag(10)
ACF_Mkt_RF=sm.tsa.acf(factor_df.iloc[0],nlags=10)
ACF_SMB=sm.tsa.acf(factor_df.iloc[1],nlags=10)
ACF_HML=sm.tsa.acf(factor_df.iloc[2],nlags=10)
ACF_RF=sm.tsa.acf(factor_df.iloc[3],nlags=10)
ACF_Mom=sm.tsa.acf(factor_df.iloc[4],nlags=10)
data_ACF=[ACF_Mkt_RF,ACF_SMB,ACF_HML,ACF_RF,ACF_Mom]
ACF_df=pd.DataFrame(data_ACF,index=['ACF_Mkt_RF','ACF_SMB','ACF_HML','ACF_RF','ACF_M
ACF_df.T
```

```
C:\Users\okanc\anaconda3\lib\site-packages\statsmodels\tsa\stattools.py:667: Future
Warning: fft=True will become the default after the release of the 0.12 release of
statsmodels. To suppress this warning, explicitly set fft=False.
  warnings.warn(
```

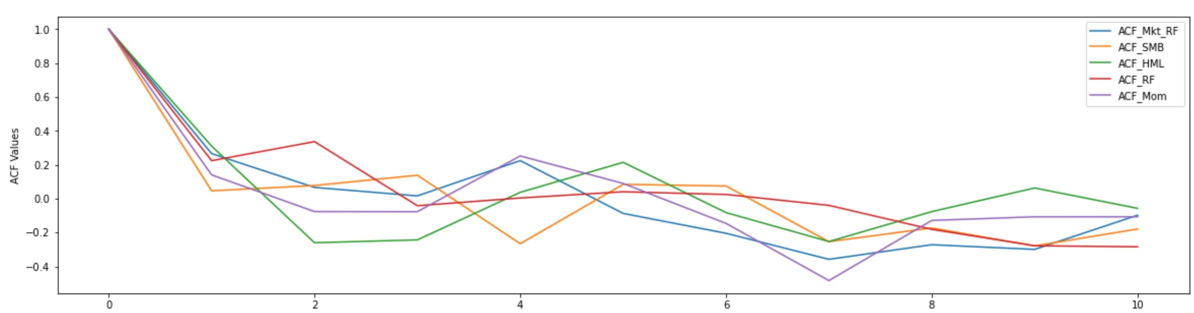
Out[6]:

	ACF_Mkt_RF	ACF_SMB	ACF_HML	ACF_RF	ACF_Mom
0	1.000000	1.000000	1.000000	1.000000	1.000000
1	0.265962	0.046341	0.309947	0.224183	0.140254
2	0.066292	0.077607	-0.259990	0.336513	-0.076050
3	0.016351	0.138315	-0.243149	-0.041576	-0.076769
4	0.224533	-0.265698	0.037447	0.004009	0.252086
5	-0.086949	0.084392	0.214473	0.040677	0.089861
6	-0.204150	0.074719	-0.082180	0.024825	-0.145821
7	-0.357957	-0.253680	-0.252920	-0.039545	-0.483255
8	-0.271716	-0.172884	-0.076457	-0.181050	-0.128542
9	-0.299380	-0.278789	0.063098	-0.277822	-0.106833
10	-0.097600	-0.179083	-0.057160	-0.284137	-0.106906

In [7]:
```
fig, ax = plt.subplots(1, 1, figsize=(20, 5))

ACF_df.T.plot(table=False, ax=ax);
plt.xlabel('Lags')
plt.ylabel('ACF Values')
```

Out[7]:
```
Text(0, 0.5, 'ACF Values')
```



As we can see on the graph that all factors have decreasing trend in terms of ACF.

Question 2:

$$X_t = 0.01 + 0.2\, X_{t-2} + a_t$$

where a_t is a Gaussian white noise series with mean zero and variance 0,02.

(a) What are the mean and variance of the return series X_t?

$$E(X_t) = E(0.01 + 0.2 X_{t-2} + a_t)$$
$$= 0.01 + E(0.2 X_{t-2}) + E(a_t)$$
$$E(a_t) = 0$$
$$\text{Stationary} \longrightarrow E(X_t) = E(X_{t-2}) = \mu$$
$$\mu = 0.01 + 0.2\mu$$
$$0,8\mu = 0,01$$

$$\text{Mean} \Rightarrow \boxed{\mu = 0,0125}$$

$$X_t = a_0 + a_1 X_{t-1} + a_2 X_{t-2} + \varepsilon_t$$
$$\mu = a_0 + \mu a_1 + \mu a_2$$
$$\mu = \frac{a_0}{1-a_1-a_2}$$

$$X_t = \mu(1-a_1-a_2) + a_1 X_{t-1} + a_2 X_{t-2} + a_t$$

$$(*) \rightarrow X_t - \mu = a_1(X_{t-1} - \mu) - a_2(X_{t-2} - \mu) + \varepsilon_t$$

$$Var(X_t) = a_1^2\, Var(X_{t-1}) - a_2^2\, Var(X_{t-2}) + \sigma_\varepsilon^2$$

Stationary Cond:

$$Var(X_t) = Var(X_{t-1}) = Var(X_{t-2})$$
$$Var(X_t) - a_1^2\, Var(X_{t-1}) - a_2^2\, Var(X_{t-2}) = \sigma_\varepsilon^2$$
$$Var(X_t)(1 - a_1^2 - a_2^2) = \sigma_\varepsilon^2 \qquad \boxed{Var(X_t) = \frac{\sigma_\varepsilon^2}{1-a_1^2-a_2^2}}$$

$$\mathrm{Var}(X_t) = \frac{0.02}{1-0-(0.2)^2} = \underline{\underline{0.02083}}$$

(b) Compute the lag-1 and lag-2 autocorrelation of X_t

Multiplying Eq. ★ by $(X_{t-\ell}-\mu)$

$$(X_{t-\ell}-\mu)\cdot(X_t-\mu) = a_1(X_{t-1}-\mu)\cdot(X_{t-\ell}-\mu) + a_2\cdot(X_{t-2}-\mu)\cdot(X_{t-\ell}-\mu)$$
$$+ \varepsilon_t\cdot(X_{t-\ell}-\mu)$$

Taking Expectation:

$$E((X_{t-\ell}-\mu)\cdot(X_t-\mu)) = a_1\cdot E((X_{t-1}-\mu)\cdot(X_{t-\ell}-\mu))$$
$$+ a_2\cdot E((X_{t-2}-\mu)(X_{t-\ell}-\mu))$$
$$+ E(\varepsilon_t\cdot(X_{t-\ell}-\mu))$$

$$\mathrm{Cov}(X_t, X_{t-\ell}) = a_1\cdot \mathrm{Cov}(X_{t-1}, X_{t-\ell}) + a_2\cdot \mathrm{Cov}(X_{t-2}, X_{t-\ell})$$
$$+ E(\varepsilon_t(X_{t-\ell}-\mu))$$

$$V_\ell = a_1 V_{\ell-1} + a_2\cdot V_{\ell-2} + 0 \quad \rightarrow r_\ell = \frac{V_\ell}{V_0} = a_1 r_{\ell-1}$$

$$r_\ell = a_1 r_{\ell-1} + a_2 r_{\ell-2} \qquad \ell \geq 0$$

for lag-1:

$$r_1 = a_1\cdot r_0 + a_2 r_{-1} \qquad r_0 = 1 \qquad a_1 = 0$$
$$r_1 = a_1 + a_2 r_{-1} \longrightarrow r_1 = r_{-1}$$
$$r_1 - a_2 r_1 = a_1$$
$$r_1 = \frac{a_1}{1-a_2} = \frac{0}{1-0.2} = 0 \qquad \boxed{r_1 = 0}$$

for lag-2:

$$r_2 = a_1\cdot r_1 + a_2\cdot r_0$$
$$r_2 = 0 + 0.2\cdot 1 \qquad \boxed{r_2 = 0.2}$$

(c) Assume that $X_{100} = -0.01$ and $X_{99} = 0.02$. Compute the 1- and 2-step-ahead forecast of the return series at the forecast origin $t=100$

$$X_t = a_0 + a_1 X_{t-1} + a_2 X_{t-2} + \varepsilon_t$$

$t = 101$

$$X_{101} = a_0 + a_1 \cdot X_{100} + a_2 X_{99} + \varepsilon_t$$

$$X_{101} = 0.01 + 0 \cdot (-0.01) + 0.2 \cdot (0.02) + \varepsilon_t$$

$$X_{101} = \underline{\underline{0.014 + \varepsilon_t}}$$

$$X_{102} = a_0 + a_1 X_{101} + a_2 X_{100} + \varepsilon_t$$

$$X_{102} = 0.01 + 0 \cdot (0.014 + \varepsilon_t) + 0.2 \cdot (-0.01) + \varepsilon_t$$

$$X_{102} = 8 \cdot 10^{-3} + \varepsilon_t$$