Principal Component Analysis for IFRS9 Forward-Looking Modeling

Andrija Djurovic

www.linkedin.com/in/andrija-djurovic

Principal Component Analysis

- Principal Component Analysis (PCA) enables practitioners to efficiently capture essential patterns and variability in the data by transforming the original variables into a smaller set of uncorrelated principal components.
- After reducing the data dimensionality, the selected principal components typically serve as inputs for the regression model.
- In the context of IFRS9 forward-looking modeling, PCA is a notable approach practitioners employ.
- PCA addresses a significant challenge in forward-looking modeling exercises: the relatively low ratio between the number of observations and the number of independent variables.
- Despite its popularity, is PCA always the optimal solution?
- Practitioners should avoid unthinkingly applying PCA for IFRS9 forward-looking modeling.
- Special consideration is essential for evaluating the observed directional relationship between the variables comprising the selected PCA and the target variable and comparing it to the anticipated directional relationship.

Case Study - Setup

1. The IFRS9 PD modeling dataset comprises four variables collected with year-on-year change: Observed Default Rate (`odr`), Gross Domestic Product (`gdp`), Unemployment (`unemployment`), and Wages (`wage`).
2. The observed default rate is the target variable, while the other variables are considered independent.
3. The dataset is structured quarterly, covering the most recent 50 quarters of the observed indicators.
4. The expected directional relationship between the independent variables and the target is assumed to be negative for `gdp` and `wage` while positive for `unemployment`. A negative directional relationship implies that as the independent variable increases, the target decreases, and vice versa. The positive directional relationship implies the same directional change between the independent and dependent variable.
5. Suppose we set a threshold of 20 observations per independent variable when selecting the maximum number of variables for the regression model. This limitation would allow us to utilize only two independent variables.
6. To incorporate all three independent variables in the model, we will transform them using PCA and then use only the first principal component as an input for the OLS regression.
7. The case study's ultimate goal is to evaluate the alignment between the anticipated and observed directional relationships among the raw independent variables after running the PCA and the target variable.

Case Study - R Code

```r
#data import
url <- "https://raw.githubusercontent.com/andrija-djurovic/adsfcr/main/model_dev_and_vld/pca.csv"
db <- read.csv(file = url,
               header = TRUE)
#expected directional relationship
dr.e <- c("gdp" = "-", "unemployment" = "+", "wage" = "-")
#pca
pca.res <- prcomp(x = db[, -1],
                  scale = TRUE)
#pca rotation
pca.res$rotation
```

```
##                     PC1         PC2        PC3
## gdp          -0.6057804  0.49897450 -0.6197213
## unemployment -0.4113107 -0.86314827 -0.2929139
## wage         -0.6810680  0.07745652  0.7281119
```

```r
#extract pc1
db$pca.1 <- pca.res$x[, "PC1"]
```

Case Study - R Code cont.

```
#ols - principal component
ols.p <- lm(formula = odr ~ pca.1,
            data = db)
ols.p
```

```
##
## Call:
## lm(formula = odr ~ pca.1, data = db)
##
## Coefficients:
## (Intercept)        pca.1
##    -0.02026      0.19604
```

```
#observed directional relationship
dr.o <- sign(coef(ols.p)[2]*pca.res$rotation[, "PC1"])
dr.o <- ifelse(dr.o == 1, "+", "-")

#compare the expected and observed directional relationship
dr.c <- data.frame(EXPECTED = dr.e,
                   OBSERVED = dr.o)
dr.c
```

```
##              EXPECTED OBSERVED
## gdp                 -        -
## unemployment        +        -
## wage                -        -
```

Case Study - Python Code

```
import pandas as pd
import numpy as np
from sklearn.decomposition import PCA
import statsmodels.formula.api as smf

#data import
fp = "https://raw.githubusercontent.com/andrija-djurovic/adsfcr/main/model_dev_and_vld/pca.csv"
db = pd.read_csv(filepath_or_buffer = fp)

#expected directional relationship
dr_e = {"gdp": "-", "unemployment": "+", "wage": "-"}
#standardize pca inputs
mv = ["gdp", "unemployment", "wage"]
db[mv] = (db[mv] - np.mean(db[mv], axis = 0)) / np.std(db[mv], axis = 0, ddof = 1)
#pca
pca = PCA(svd_solver = "full")
pca_res = pca.fit_transform(X = db.iloc[:, 1:])
#pca rotations
pca.components_
```

```
## array([[-0.60578043, -0.41131073, -0.68106795],
##        [-0.4989745 ,  0.86314827, -0.07745652],
##        [-0.61972132, -0.2929139 ,  0.7281119 ]])
```

```
#extract pc1
db["pca_1"] = pca_res[:, 0]
```

Case Study - Python Code cont.

```
#ols - principal component
ols_p = smf.ols(formula = "odr ~ pca_1",
                data = db).fit()
ols_p.params
```

```
## Intercept   -0.020262
## pca_1        0.196038
## dtype: float64
```

```
#observed directional relationship
dr_o = np.sign(ols_p.params["pca_1"]* pca.components_[0])
dr_o = np.where(dr_o == 1, "+", "-")

#compare the expected and observed directional relationship
dr_c = pd.DataFrame({"EXPECTED": dr_e.values(),
                     "OBSERVED": dr_o},
                     index = dr_e.keys())
dr_c
```

```
##              EXPECTED OBSERVED
## gdp                 -        -
## unemployment        +        -
## wage                -        -
```

Discussion Points

- What adjustments are necessary when employing more than one principal component?
- How does the integration of time lags for independent variables add complexity to the PCA procedure?
- What are the alternatives to the PCA to address the same challenge in the IFRS9 forward-looking modeling?
- How can PCA be applied in other credit risk domains, such as IRB model development or scorecard modeling?