Computation and Application to Dataset

Dataset 1: Walmart Store Data from Kaggle

From a 2014 Kaggle competition with a goal of forecasting Walmart Sales, the Walmart Store dataset provides information on sales for $n = 45$ located in different regions of the United States. Each store has Weekly Sales data from $2010 - 02 - 05$ to $2012 - 11 - 01$, information by department and a binary IsHoliday variable. Evidently, this data has fairly limited number of predictors but was one of few available retail datasets, thus was used as a warm up application before applying to a larger dataset.

[@walmartdata]

Prior Analysis : Kimmo Kiviluoto and Erkki Oja

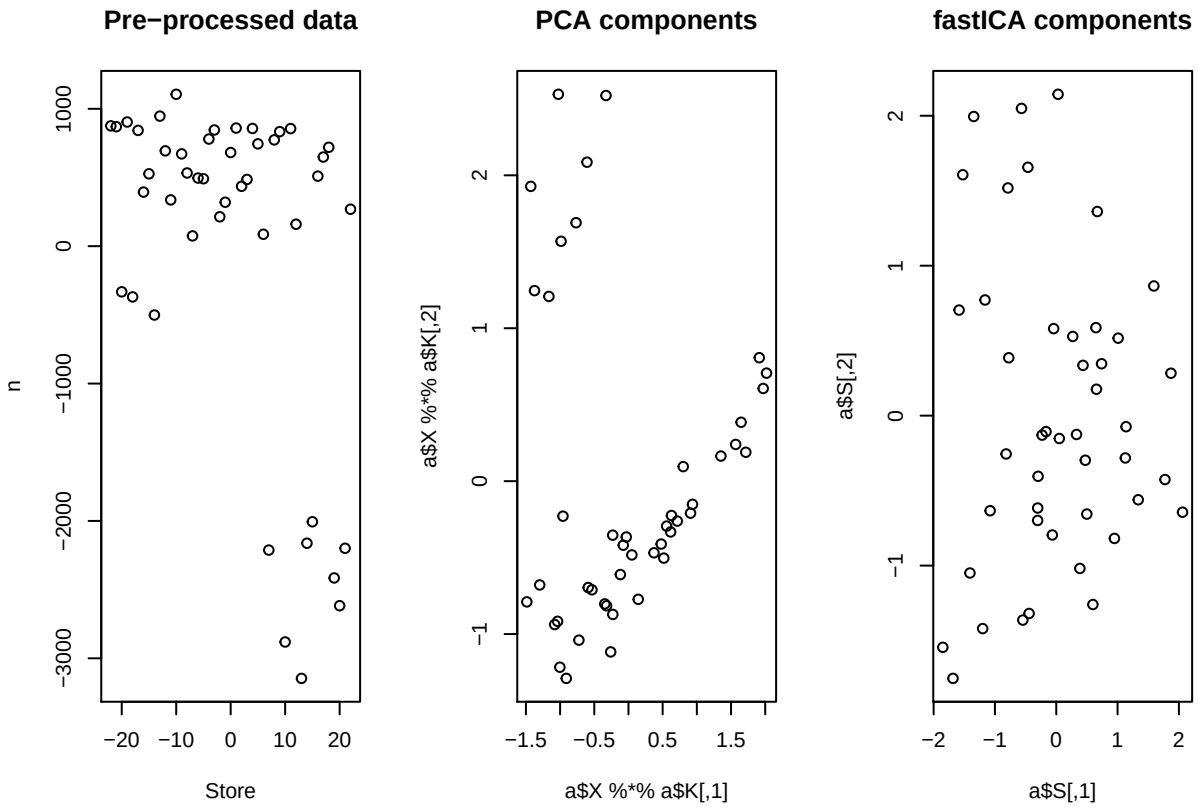
In 1998, Kiviluoto and Erkki Oja used ICA on parallel Financial Time Series from a retail chain of 40 stores. They claim in their paper *Independent Component Analysis for Parallel Financial Time Series* that by removing the fundamental factors of the data, they were able to see the impact of management decisions of a particular store more clearly. [pg. 2,@kimmo1998]

With the Walmart store data, an attempt to show a similar result using retail data will be made. However, this will look at Weekly Sales not Cash-flow data, which will probably not generate as robust results since cash-flow provides more information about financial activities of a store than just sales.

Exploration

FastICA Application

```
# Set nc= 3 based on dimensions
a <-fastICA(w2,3, alg.typ="parallel",fun= "logcosh",row.norm=FALSE,maxit=5,tol= 0.0001,verbose=TRUE)
par(mfrow = c(1, 3))
plot(a$X, main = "Pre-processed data")
plot(a$X %*% a$K, main = "PCA components" )
plot(a$S, main = "fastICA components")
```



From these plots, there appears to be some clustering uncovered by the fastICA algorithm. One possibility is that the cluster of 8 stores at towards the top represents the most productive in terms of sales but this could just be random noise. In general,the dataset appears to be too small in terms of both dimensions and observations to come to any explicit conclusions about underlying influences of Weekly Sales and top performing stores.

Dataset 2: Sample Superstore data

Sample data that appears in the December Tableau User Group presentation. Since this data has higher dimensionality and more variety of variables, the ICA model should be more effective here. Unlike with the prior simulations, only the fastICA algorithm will be used. [@storedata]

Note: Geographic locations have been altered to include Canadian locations numerically (provinces / regions).

Select Variables of Interest

```
## # A tibble: 21 x 1
##                  x
##              <chr>
##  1         Row ID
##  2       Order ID
##  3     Order Date
##  4 Order Priority
##  5 Order Quantity
##  6          Sales
##  7       Discount
##  8      Ship Mode
```

```
##  9         Profit
## 10     Unit Price
## # ... with 11 more rows
```

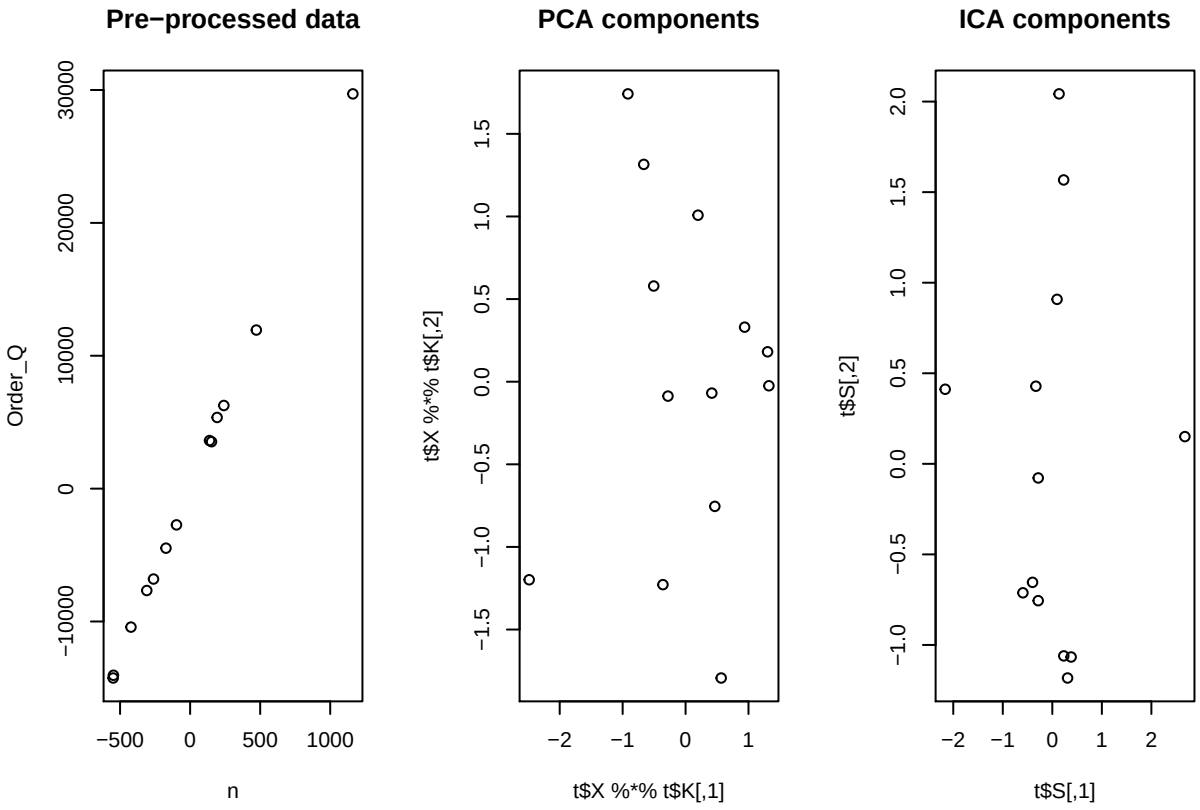
By Province

```
s <- store %>%
  select(`Order ID`,`Row ID`,`Order Date`,`Order Quantity`,Sales, Discount, Profit,`Unit Price`, `Shipp
  group_by(Province) %>% summarise(n = n(),
    Order_Q = sum(`Order Quantity`),
    Sales = sum(Sales),
    Discount = sum(Discount),
    Profit = sum(Profit),
    U_Price = sum(`Unit Price`),
    Ship_C = sum(`Shipping Cost`))

x <- s[,-1]
```

```
t <- fastICA(x,4, alg.typ="parallel",fun= "logcosh",row.norm=FALSE,maxit=5,tol= 0.0001,verbose=TRUE)
```

```
par(mfrow = c(1, 3))
plot(t$X, main = "Pre-processed data")
plot(t$X %*% t$K, main = "PCA components" )
plot(t$S, main = "ICA components")
```

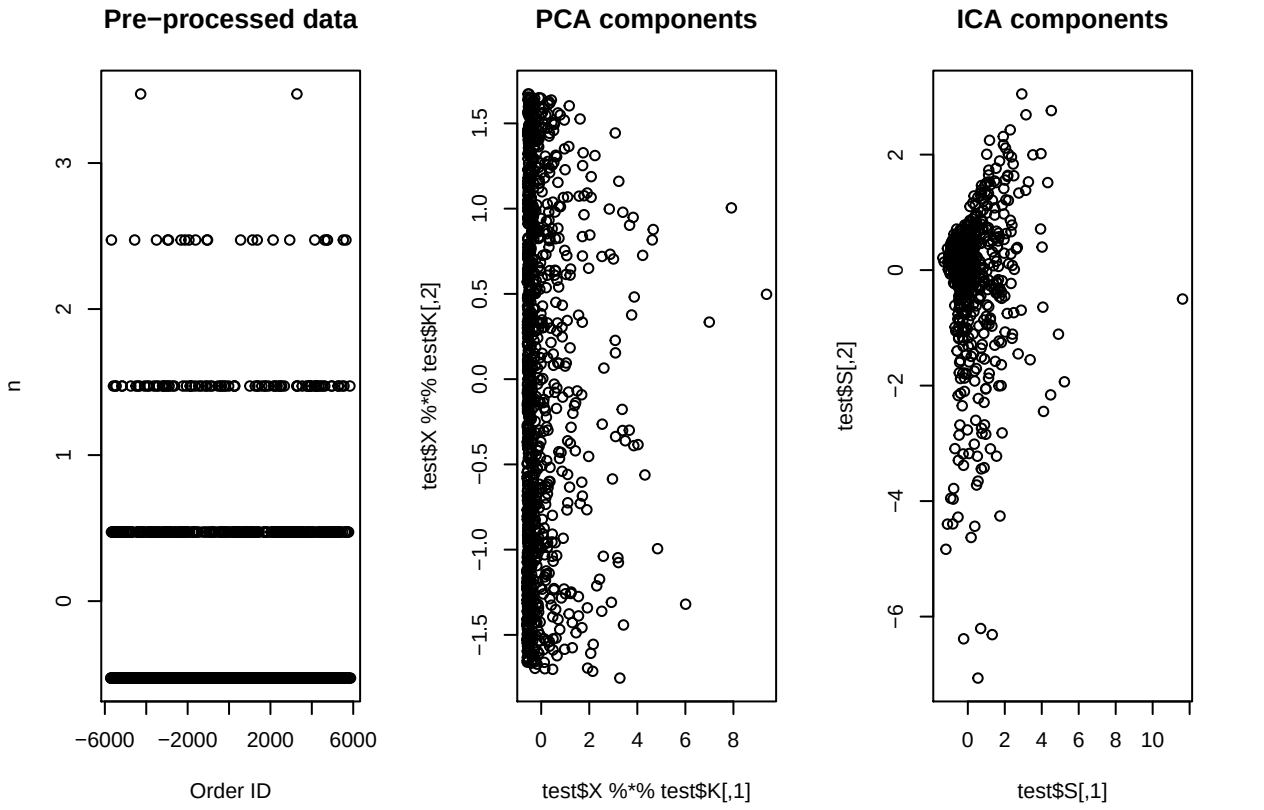


Looking at the superstore data by Province, some limited clustering exists that could potentially indicate nuances in store performance by Province or more volume of orders. However, this view does not seem to be robust enough for any substantial conclusions.

By Order ID

```
z <- store %>%
  select(`Order ID`,`Row ID`,`Order Date`,`Order Quantity`,Sales, Discount, Profit,`Unit Price`, `Shipp
  group_by(`Order ID`) %>% summarise(
  n = n(),
    Order_Q = sum(`Order Quantity`),
    Sales = sum(Sales),
    Discount = sum(Discount),
    Profit = sum(Profit),
    U_Price = sum(`Unit Price`),
    Ship_C = sum(`Shipping Cost`))
z1 <- z[sample(1000),] #take sub sample
```

```
test <- fastICA(z1,8, alg.typ="parallel",fun= "logcosh",row.norm=FALSE,maxit=5,tol= 0.0001,verbose=TRUE)
par(mfrow = c(1, 3))
plot(test$X, main = "Pre-processed data")
plot(test$X %*% test$K, main = "PCA components" )
plot(test$S, main = "ICA components")
```



While the PCA model provides minimal information, the ICA model for the superstore data by order ID appears to have separated some particular ID's from a massive cluster of orders. These orders could be the most atypical in terms of quantity or price, but this separation indicates the effectiveness of the ICA model on the data in general.

Run 1000 fastICA Iterations

```
set.seed(10)
N <- 1000
itF <- rep(NA,1,N)

colMax <- function(data) sapply(data, max, na.rm = TRUE)[1]
gen_s <- function(N,S)  {
  for (i in 1:N) {
    test <- fastICA(z1,8, alg.typ="parallel",fun= "logcosh",row.norm=FALSE,maxit=5,tol= 0.0001,verbose="

    itF[i] <- colMax(abs(congru(S,test$S)))


  }
  return(itF)
}
```

Absolute Maxmimum Congruency Coefficient

```
kable(favstats(reps))
```

min	Q1	median	Q3	max	mean	sd	n	missing
4.17e-05	0.0069342	0.0172902	0.0355768	0.5157638	0.074891	0.1511403	1000	0

Like with the CPP simulation, the superstore data had to be compared to the observations as opposed to the true S. Unsurprisingly, this produces a very low with high standard deviation so unfortunately does not inform the analysis in a particularly useful way.

Largely, these two dataset applications may have not been the most compelling examples of the power of ICA due to their limited dimensionality and lack of complexity.