Data Analysis of IBM Attrition

IBM 사조

공준택 김순영 김혜린 오희준 장인아

CONTENTS

01 Previously

01 Previously

	A	B	C	D	E	F	G	H
1	Age	Attrition	BusinessTr	DailyRate	Departme	DistanceFr	Education	Education
2	41	Yes	Travel_Rar	1102	Sales	1	2	Life Scienc
3	49	No	Travel_Fre	279	Research &	8	1	Life Scienc
4	37	Yes	Travel_Rar	1373	Research &	2	2	Other
5	33	No	Travel_Fre	1392	Research &	3	4	Life Scienc
6	27	No	Travel_Rar	591	Research &	2	1	Medical
7	32	No	Travel_Fre	1005	Research &	2	2	Life Scienc
8	59	No	Travel_Rar	1324	Research &	3	3	Medical
9	30	No	Travel_Rar	1358	Research &	24	1	Life Scienc
10	38	No	Travel_Fre	216	Research &	23	3	Life Scienc
11	36	No	Travel_Rar	1299	Research &	27	3	Medical
12	35	No	Travel_Rar	809	Research &	16	3	Medical
13	29	No	Travel_Rar	153	Research &	15	2	Life Scienc
14	31	No	Travel_Rar	670	Research &	26	1	Life Scienc
15	34	No	Travel_Rar	1346	Research &	19	2	Medical
16	28	Yes	Travel_Rar	103	Research &	24	3	Life Scienc
17	29	No	Travel_Rar	1389	Research &	21	4	Life Scienc
18	32	No	Travel_Rar	334	Research &	5	2	Life Scienc
19	22	No	Non-Trave	1123	Research &	16	2	Medical

Data set:

IBM 이직 데이터

주제 :

IBM 직원들의 이직 결정 **요인**들을 분석하여
지나친 이직을 막기위한 방법을 **제시**

FIN

01 Previously

Attrition	이직 여부

+

Relationship Satisfaction	관계 만족도
TotalWorkingYears	총 근무 기간(년)
TrainingTimesLastYear	작년 직업 훈련시간
WorkLifeBalance	일과 생활의 균형

Yes (1233)

No (237)

34개 설명변수

FIN

- Logistic Reg.

1. step : 단계적 알고리즘을 사용해 AIC를 기준으로 모델 선택

```
null<-glm(Attrition~1,data=dat,family="binomial")
full<-glm(Attrition~.,data=dat,family="binomial")
step(null,scope = list(lower=null,upper=full),direction = "both")
```

변수의 추가와 삭제를 반복한 결정된 최종 모델

```
> formula(step1)
Attrition ~ OverTime + JobRole + MaritalStatus + EnvironmentSatisfaction +
    JobSatisfaction + JobInvolvement + BusinessTravel + YearsInCurrentRole +
    YearsSinceLastPromotion + DistanceFromHome + NumCompaniesWorked +
    Age + WorkLifeBalance + RelationshipSatisfaction + TrainingTimesLastYear +
    YearsWithCurrManager + Gender + EducationField + TotalWorkingYears +
    YearsAtCompany + StockOptionLevel
```

- Logistic Reg.

2. glm : 선택된 변수들로 적합

```
> fit<-glm(formula = Attrition ~ OverTime + TotalWorkingYears + MaritalStatus +
+              EnvironmentSatisfaction + JobInvolvement + JobSatisfaction +
+              NumCompaniesWorked + DistanceFromHome + YearsSinceLastPromotion +
+              YearsInCurrentRole + RelationshipSatisfaction + WorkLifeBalance +
+              Age + Gender + MonthlyIncome + Department + YearsWithCurrManager +
+              YearsAtCompany + TrainingTimesLastYear + StockOptionLevel +
+              DailyRate, family = "binomial", data = dat)
> summary(fit)

Call:
glm(formula = Attrition ~ OverTime + TotalWorkingYears + MaritalStatus +
    EnvironmentSatisfaction + JobInvolvement + JobSatisfaction +
    NumCompaniesWorked + DistanceFromHome + YearsSinceLastPromotion +
    YearsInCurrentRole + RelationshipSatisfaction + WorkLifeBalance +
    Age + Gender + MonthlyIncome + Department + YearsWithCurrManager +
    YearsAtCompany + TrainingTimesLastYear + StockOptionLevel +
    DailyRate, family = "binomial", data = dat)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.7151  -0.5395  -0.2941  -0.1203   3.4094
```

FIN

- Logistic Reg.

+

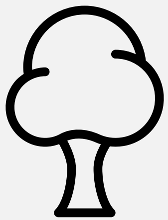
OverTime (초과 근무)
MaritalStatus (결혼 상태)
BusinessTravel (출장)
NumCompaniesWorked (일했던 직장 수)
DistanceFromHome (집과의 거리)
YearsSinceLastPromotion (승진 후 경과 시간)

−

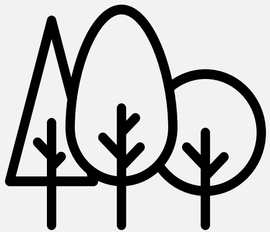
TotalWorkingYears (총 근무 기간)
EnvironmentSatiscation (환경 만족도)
JobInvolvement((직업 소속감)
JobSatisfaction (직업 만족도)
YearsInCurrentRole (직무 연차)
RelationshipSatisfaction (관계 만족도)

01 Previously

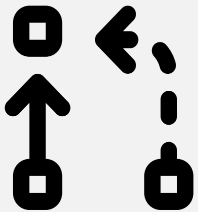
- This week



Decision Tree



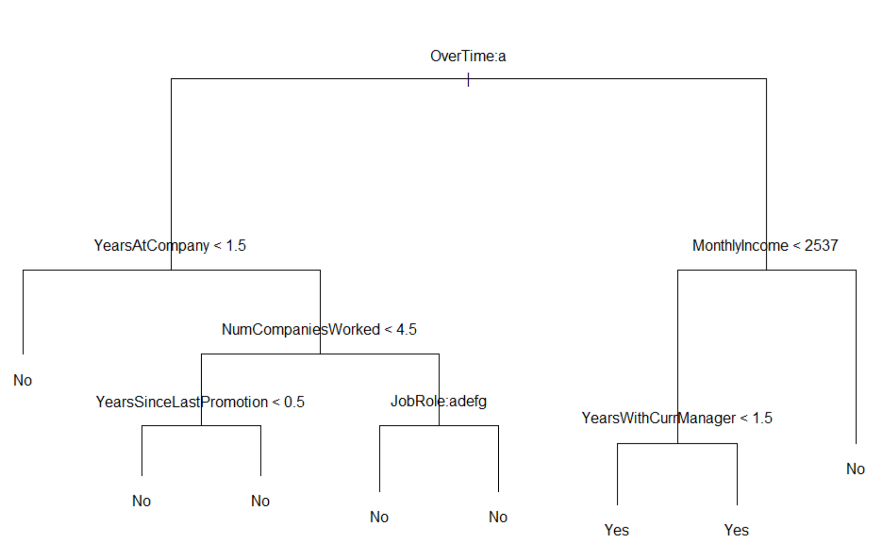
Random Forest



SVM

02 Decision Tree

<Original Data>




```
> roc.curve(test$Attrition, pred.tree.ori[,2])
Area under the curve (AUC): 0.687
```

한 쪽으로 편향된 결과

```
> table(dat$Attrition)

  No  Yes
1233  237
```

(No가 yes보다 10배 정도 많음)

한 쪽으로 편향된 데이터를 어떻게 다룰 것인가?

1. Oversampling
2. Undersampling
3. Synthetic Data Generation

('ROSE' package 사용)
Random Over Sampling Examples

1. Oversampling

Minority class의 데이터를 replicate → balance를 맞춤

- Information loss가 없음
- Overfitting의 문제 (∵ 같은 obs.가 여러 존재)

```
data.over <- ovun.sample(Attrition~., data = train, method="over",N=table(train$Attrition)[1]*2)$data
```

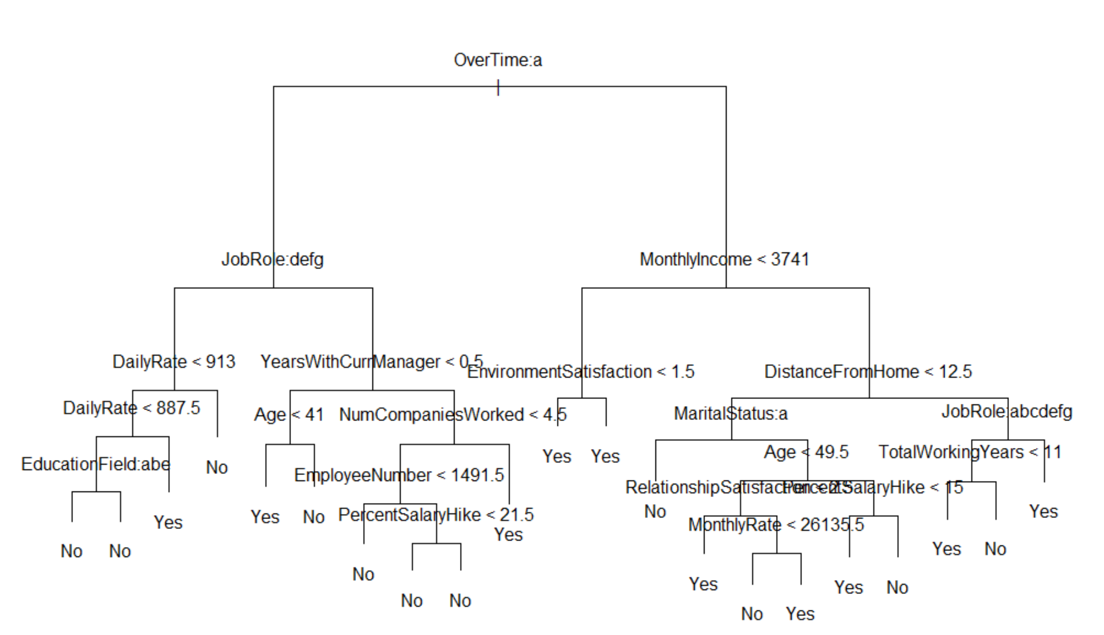
(train data에서)
yes가 no의 개수만큼 되도록

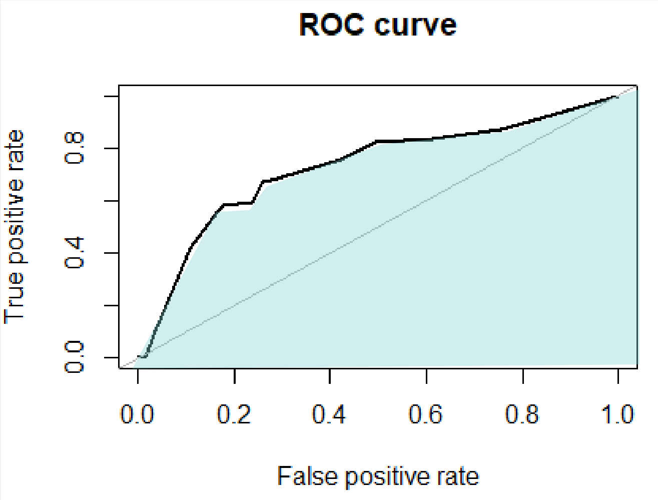
```
> table(train$Attrition)

 No Yes
871 158
```

→

```
> table(data.over$Attrition)

 No Yes
871 871
```

<완전 성장 tree> 가지치기 할 수 없었음






```
> roc.curve(test$Attrition, pred.tree.over[,2])
Area under the curve (AUC): 0.726
```

FIN

2. Undersampling

Majority class의 데이터를 줄임 → balance를 맞춤

- 데이터 크기가 클 때 유용 (계산 시간 감소)
- 중요한 정보의 손실이 발생할 수 있음

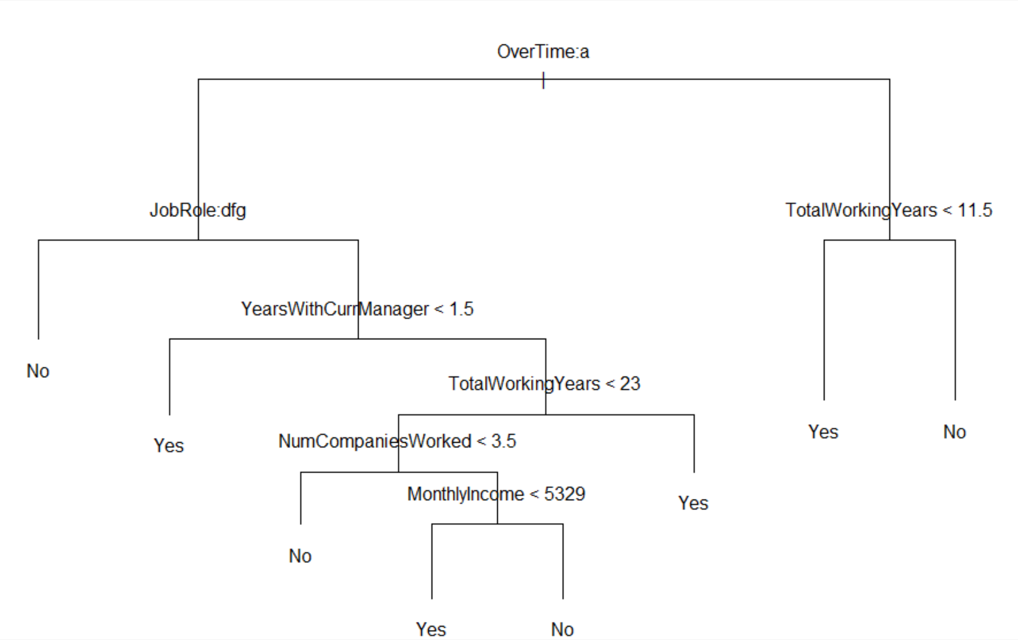
```
data.under <- ovun.sample(Attrition~.,data=train,method = "under",N=table(train$Attrition)[2]*2,seed = 1)$data
```

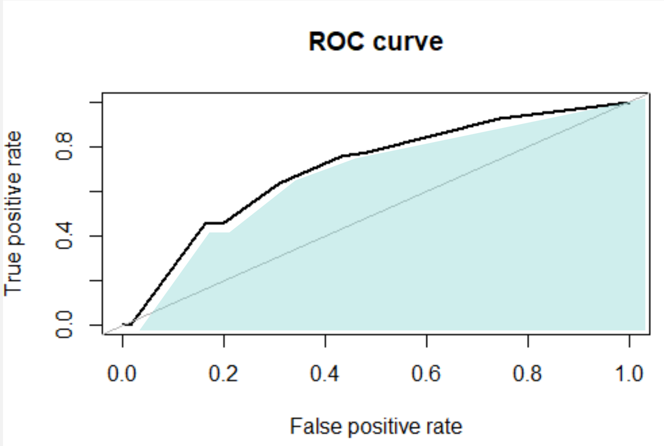
(train data에서)
no가 yes의 개수만큼 되도록

```
> table(train$Attrition)

 No Yes
871 158
```

```
> table(data.under$Attrition)

 No Yes
158 158
```

<가지치기한 tree>






```
> roc.curve(test$Attrition, pred.tree.under[,2])$auc
[1] 0.7023044
```

3. Synthetic Data Generation

데이터를 줄이거나 더하는 대신, 새로운 데이터를 만들어냄

- Synthetic minority oversampling technique (SMOTE) 이 널리 사용됨
- Bootstrapping과 k-nearest neighbors 이용

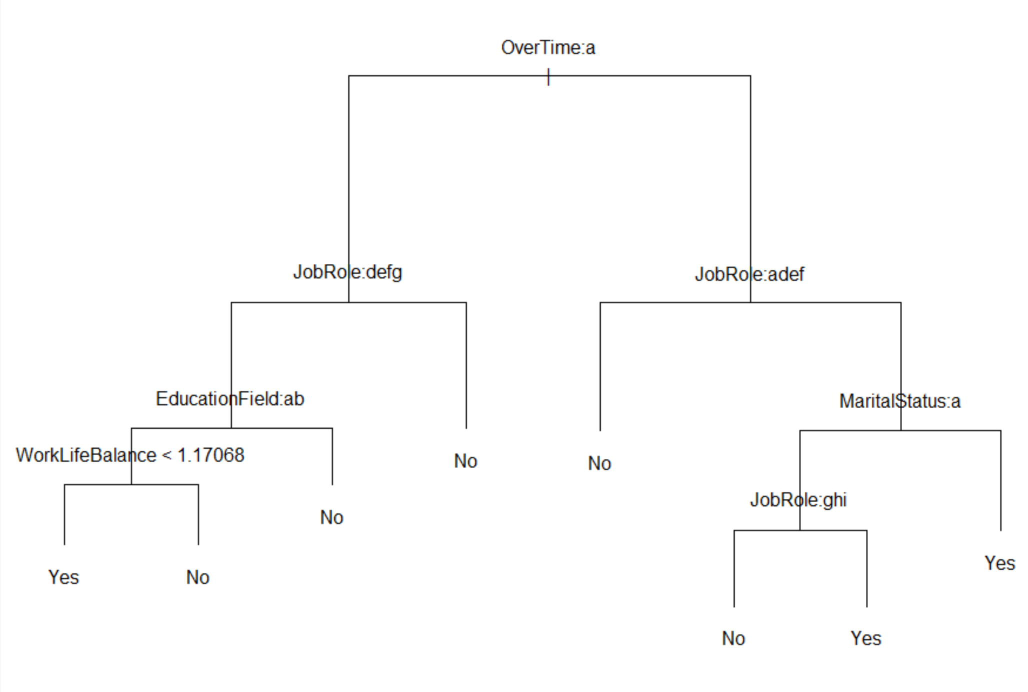
```
data.rose <- ROSE(Attrition ~ ., data = train, seed = 1)$data
```

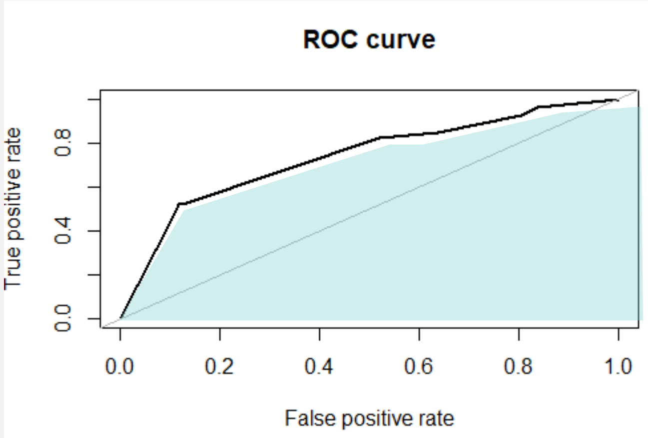
Synthetic하게 데이터 생성

```
> table(train$Attrition)

 No Yes
871 158
```

```
> table(data.rose$Attrition)

 No Yes
533 496
```

<가지치기한 tree>






```
> roc.curve(test$Attrition, pred.tree.rose[,2])
Area under the curve (AUC): 0.737
```

03 Random Forest

randomForest: 다양한 의사 결정 나무를 만들어서 학습하고, 예측 시에 여러 모델의 예측 결과들을 종합해 사용한다.

장점) 일반적으로 성능이 뛰어나고, 여러 의사 결정 나무를 사용해 과적합 문제를 피한다.

Random Forest 생성

varImpPlot 변수 중요도

Predict 예측

Auc 모델 평가

Random Forest 생성

```
library(ROSE)
library(randomForest)
data.rose <- ROSE(Attrition ~ ., data = train, seed = 1)$data
rf.fit<-randomForest(Attrition~.,data=data.rose,ntree=500,mtry=5)
```

randomForest : 랜덤 포레스트 생성

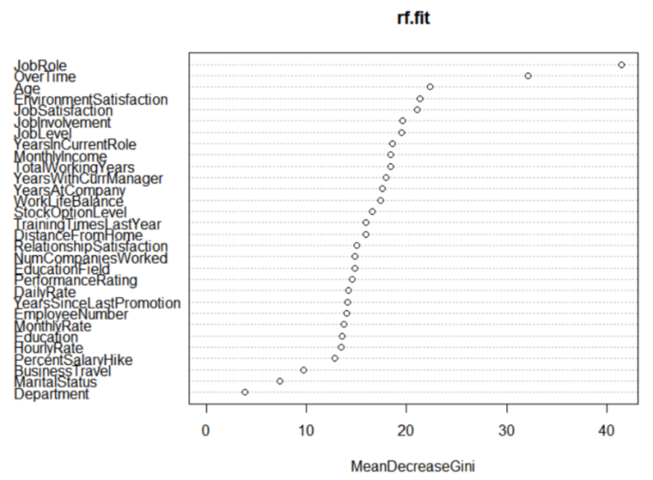
- **data.rose** : yes / no 불균형한 데이터를 비율을 맞춰준 data 사용
- **ntree** : 나무의 개수
- **mtry** : 노드 나눌 때 고려할 변수의 개수

03 Random Forest

varImp Plot

```
varImpPlot(rf.fit)
```

1) Job Role
2) Over Time
3) Age
4) Environment Satisfaction
5) Job satisfaction



varImpPlot : 변수의 중요도를 평가하고 모델링에 사용할 변수를 선택

- **importance( ) / varImpPlot( )** 사용해 결과 출력

Predict
예측

```
library(prediction)
library(ROCR)
pred.rf<-predict(rf.fit,newdata=test)
pred<-prediction(as.integer(pred.rf),test$Attrition)
```

predict : 만들어진 랜덤 포레스트에 test 데이터를 적용시켜 예측해본다.

Predict
예측

```
> auc<-performance(pred,measure = "auc")
> auc<-auc@y.values[[1]]
> auc
[1] 0.7313216
```

auc = 1.0 일 때 가장 완벽한 예측

auc = 0.7313

04 SVM

SVM
생성

```
> svm <- svm(Attrition ~.,
+            data = training_data_formula
+ )
> predictions_svm <- predict(svm, validation_data)
> summary(svm)

Call:
svm(formula = Attrition ~ ., data = training_data_formula)


Parameters:
   SVM-Type:  C-classification
 SVM-Kernel:  radial
       cost:  1
      gamma:  0.02941176

Number of Support Vectors:  291

 ( 141 150 )


Number of Classes:  2

Levels:
 No Yes
```

E1071 패키지를 사용

```
predictions_svm  No Yes
            No  317  24
            Yes  52  47
```

Parameter 조정

```
Parameter tuning of 'svm':

- sampling method: 10-fold cross validation

- best parameters:
 gamma cost
   0.1    1

- best performance: 0.2428049
```

tune.svm 함수를 통해 최적의 **gamma**와 **cost**를 찾아내 **svm**에 적용

04 Suggestion

IBM

두 SVM
비교

```
> roc.curve(validation_data$Attrition, predictions_svm)
Area under the curve (AUC): 0.761
> roc.curve(validation_data$Attrition, predictions_after_svm)
Area under the curve (AUC): 0.789
```

04 Suggestion


```
> t(svm_model_after_tune$coefs) %*%svm_model_after_tune$SV
     OverTimeNo OverTimeYes TotalWorkingYears MaritalStatusMarried MaritalStatusSingle EnvironmentSatisfaction.L
[1,]  -21.48652    21.48652         -18.71656            -3.440229            8.805802                 -12.16127
     EnvironmentSatisfaction.Q EnvironmentSatisfaction.C JobInvolvement.L JobInvolvement.Q JobInvolvement.C JobSatisfaction
[1,]                  9.698596                 -3.985678        -8.760645         3.919396        0.8491415       -6.497664
     NumCompaniesWorked DistanceFromHome YearsSinceLastPromotion YearsInCurrentRole RelationshipSatisfaction.L
[1,]           8.809673          4.09585                5.043538           -15.1861                  -1.964821
     RelationshipSatisfaction.Q RelationshipSatisfaction.C WorkLifeBalance.L WorkLifeBalance.Q WorkLifeBalance.C       Age
[1,]                    2.89335                  -6.572542         -4.388903          3.446525          2.538832 -11.91989
     GenderMale MonthlyIncome DepartmentResearch...Development DepartmentSales YearsWithCurrManager YearsAtCompany
[1,]   2.302842     -17.79618                        -4.335107        3.887916            -18.69237      -8.920574
     TrainingTimesLastYear StockOptionLevel.L StockOptionLevel.Q StockOptionLevel.C DailyRate
[1,]              -15.1428          -8.289982           13.52427          -4.088099 -7.281296
```

05 Conclusion

05 Conclusion

[분석 방법 별 중요 변수]

Decision Tree

1) Over Time

2) JobRole

3) Education Field

4) MaritalStatus

5) WorklifeBalance

Random Forest

1) Job Role

2) Over Time

3) Age

4) Environment Satisfaction

5) Job satisfaction

[conclusion] 기업은 Over time 줄여라!!

- 입사 후 결정되는 **"Over time", "Work and Life Balance"**가 이직을 막을 결정적 변수
- **Job Role, Age, Marital Stat**e 는 현재 직장과 무관하게 정해지는 변수
- Saticfaction은 전반적인 조건을 모두 통틀어 측정된다고 보아 해석x
- 임금과 관련된 변수는 초기 예상과 다르게 중요변수로 작용하지 않았음

FIN

[Appendix] 그리고 그것은 사실로 밝혀졌다…

IBM, 초과근무수당 안주다 고소당해

입력시간 | 2006. 01. 25 09:32 | 홍정민 jmhong@edaily.co.kr

IBM 직원들 초과근무 수당 청구소송

라디오코리아 | 입력 01/24/2006 12:01:00

(앵커멘트) 세계 최대 컴퓨터 회사 IBM이 수천명의 평사원…
하지 않았다는 이유로 소송을 당했습니다. 보도에 김연신 기자입니다.

컴퓨팅 |

비용절감 IBM, 계약직 초과근무 금지

김우용 기자 | 입력 : 2013.05.03.08:41 | 수정 : 2013.05.03.10:20

고소당해

jmhong@edaily.co.kr


IBM
[THANK YOU]