**프로젝트 I**

|  |  |
| --- | --- |
| 프로젝트 명 | *Project1* |
| 이 름 | *송창헌* |
| 문서 제목 | 결과보고서 |

|  |  |
| --- | --- |
| **Version** | 1.0 |
| **Date** | 28 |

|  |  |
| --- | --- |
| **팀원** | 송 창 헌 |
|  |
|  |
|  |
|  |

|  |
| --- |
| **CONFIDENTIALITY/SECURITY WARNING**  이 문서에 포함되어 있는 정보는 국민대학교 전자정보통신대학 컴퓨터공학부 및 컴퓨터공학부 개설 교과목 비주얼 컴퓨팅 최신기술 수강 학생 중 프로젝트 “프로젝트 1”를 수행하는 “송창헌”의 자산입니다. 국민대학교 컴퓨터공학부 및 “송창헌”의 서면 허락없이 사용되거나, 재가공 될 수 없습니다. |

**문서 정보 / 수정 내역**

|  |  |
| --- | --- |
| **Filename** | Project\_1.docx |
| **원안작성자** | 송창헌 |
| **수정작업자** | 송창헌 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 수정날짜 | 대표수정자 | Revision | 추가/수정 항목 | 내 용 |
| 2017-3-28 | 송창헌 | 1.0 | 작성 | 전체 내용 작성 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**목 차**

[1 개요 4](#_Toc478490553)

[1.1 프로젝트 개요 4](#_Toc478490554)

[2 개발 내용 및 결과물 5](#_Toc478490555)

[2.1 목표 5](#_Toc478490556)

[2.2 연구/개발 내용 및 결과물 5](#_Toc478490557)

[2.2.1 연구/개발 내용 5](#_Toc478490558)

[2.2.2 활용/개발된 기술 6](#_Toc478490559)

[2.2.3 결과물 목록 6](#_Toc478490560)

# 개요

## 프로젝트 개요

본 프로젝트는 비주얼 컴퓨팅 최신기술 첫번째 프로젝트로, 주어진 training 데이터를 이용하여 test 데이터를 classification 하는 것을 목적으로 한다. 각 Iris는 4가지의 featur를 가지며 3가지의 class로 구분하는 코드를 구현한다.

1) Iris의 4가지 feature를 사용하여 각 class 당 40개의 training set과 10 개의 testing set을 주고 10개의 샘플에 대한 공분산을 구하고 이후 계산을 통한 confusion matrix를 구하는 것을 목표로 한다.

2) Iris의 2가지 feature를 사용하여 각 class 당 40개의 training set과 10 개의 testing set을 주고 10개의 샘플에 대하여 공분산을 구하고 Mahalanobis 거리를 구하여 화면에 나타내는 것을 목표로 한다.

# 개발 내용 및 결과물

## 목표

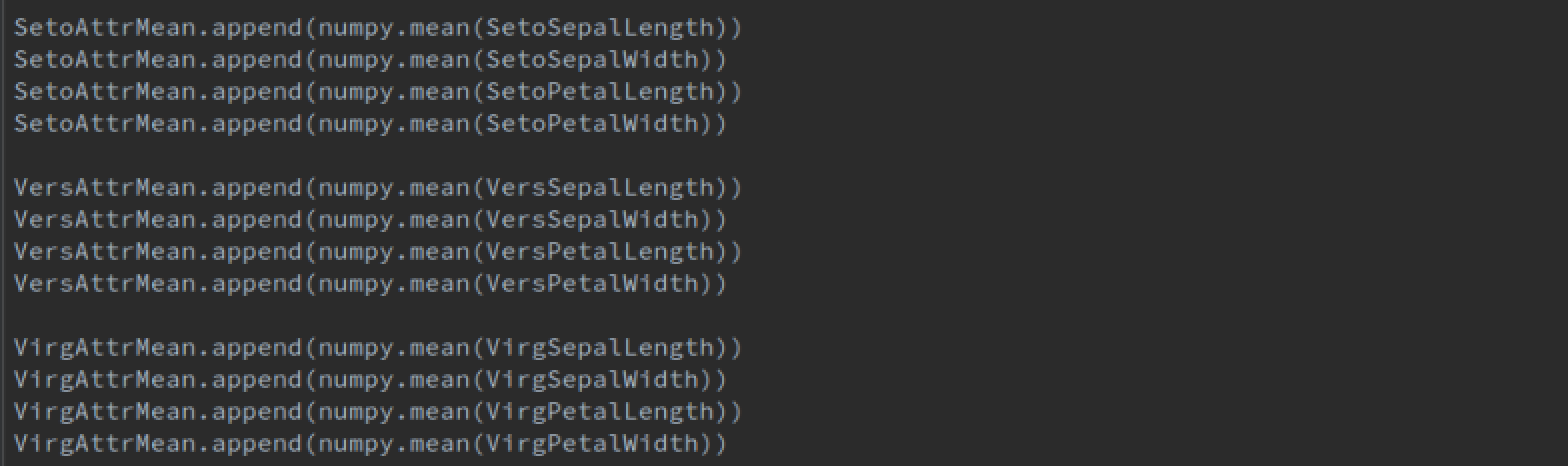
Iris의 4가지 feature를 사용하여 각 class 당 40개의 training set과 10 개의 testing set을 주고 10개의 샘플에 대한 공분산을 구하고 이후 계산을 통한 confusion matrix를 구하는 것을 목표로 한다.

Iris의 2가지 feature를 사용하여 각 class 당 40개의 training set과 10 개의 testing set을 주고 10개의 샘플에 대하여 공분산을 구하고 Mahalanobis 거리를 구하여 화면에 나타내는 것을 목표로 한다

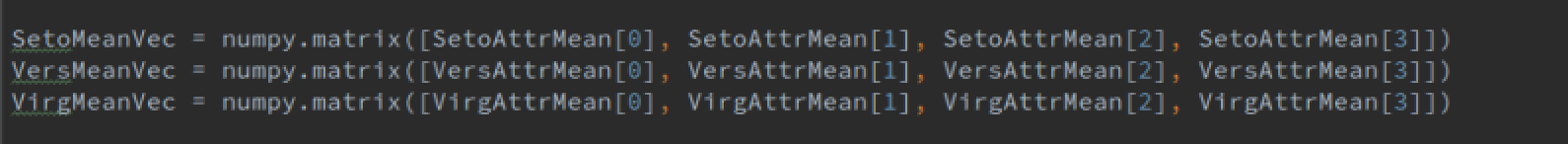
## 연구/개발 내용 및 결과물

### 연구/개발 내용

**각 class 별 구분 방식으로 공분산행렬을 이용하기 위해서, input 데이터인 “Iris\_train.dat” 에서 각각의 class마다 4개의 feature를 계산하여 벡터로 형성하고 평균값을 구한다.**

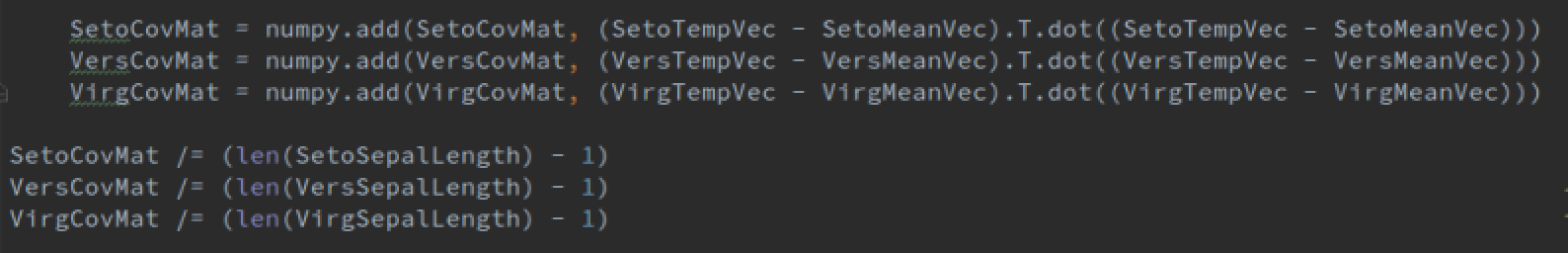


[그림 ] input data 입력



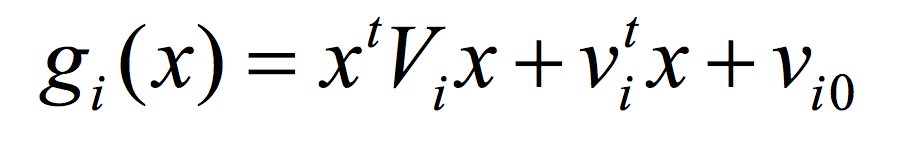
[그림 ] 평균 벡터 계산

구해진 평균 벡터를 이용하여 각 class 별 공분산 행렬을 구한다.



[그림 ] 공분산 행렬

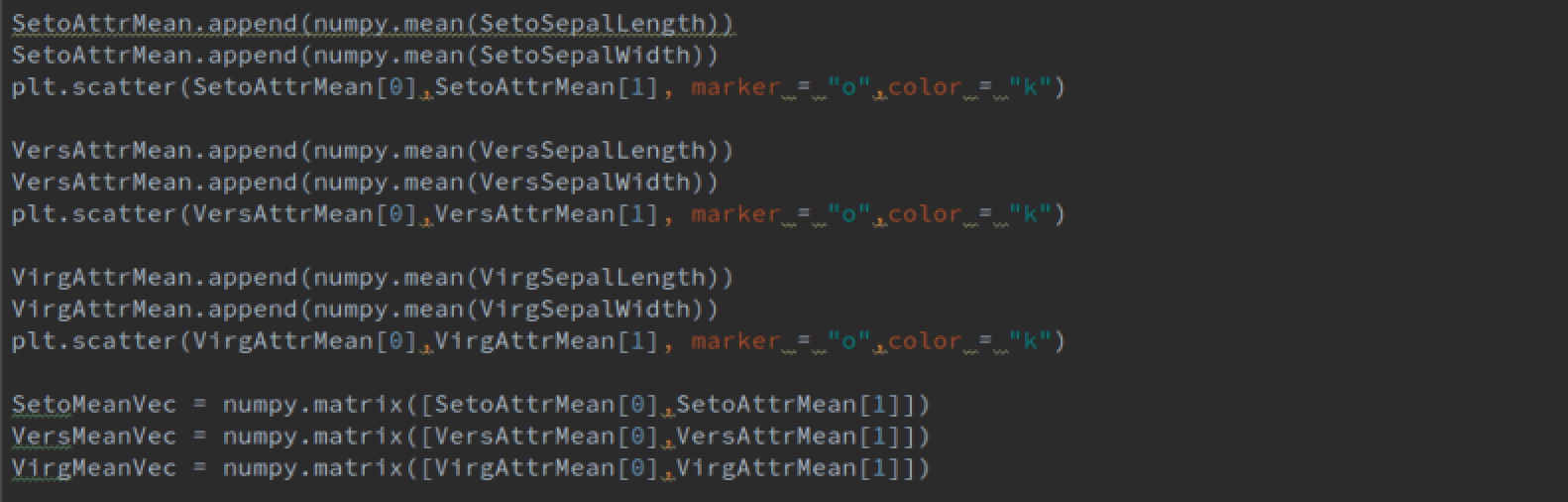
구해진 공분산 행렬과 들어오는 test data를 이용하여, 각각의 class 마다 공분산 행렬을 다른 값을 가지기 때문에 다음과 같은 식을 사용하여 decision boundaries를 구한다.



[그림 ] decision boundaries

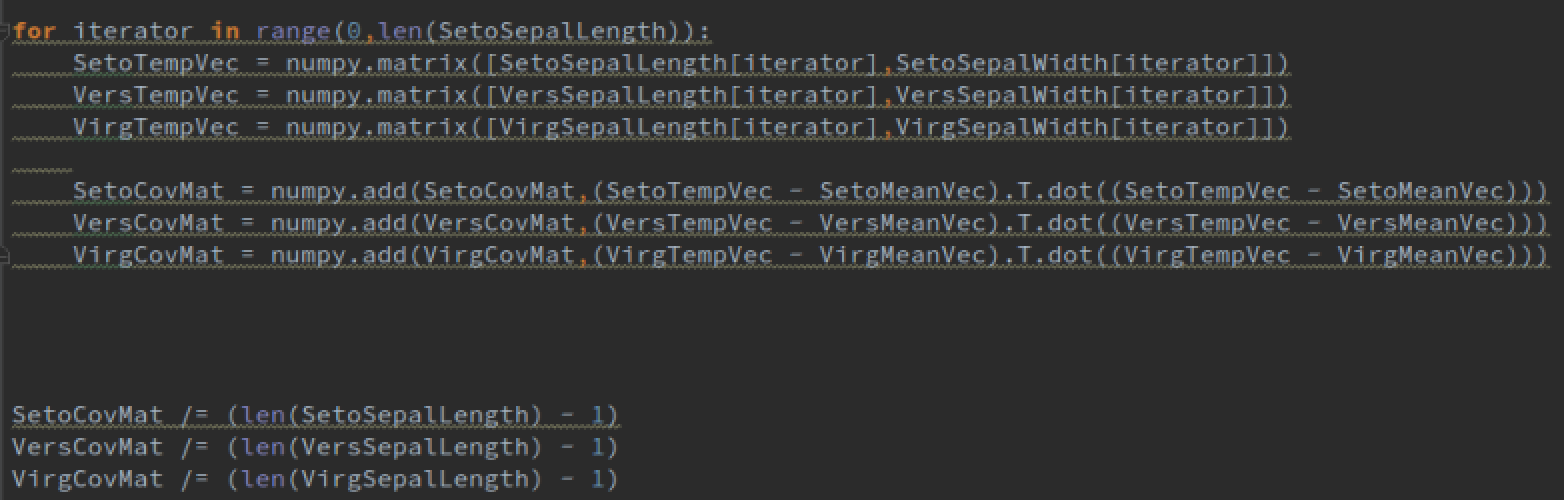
이후 값을 비교하여 confusion matrix를 완성한다.

input 데이터인 “Iris\_train.dat” 에서 각각의 class마다 4개의 feature를 계산하여 벡터로 형성하고 평균값을 구한다. Scatter를 이용하여 화면에 Training data를 표시한다.



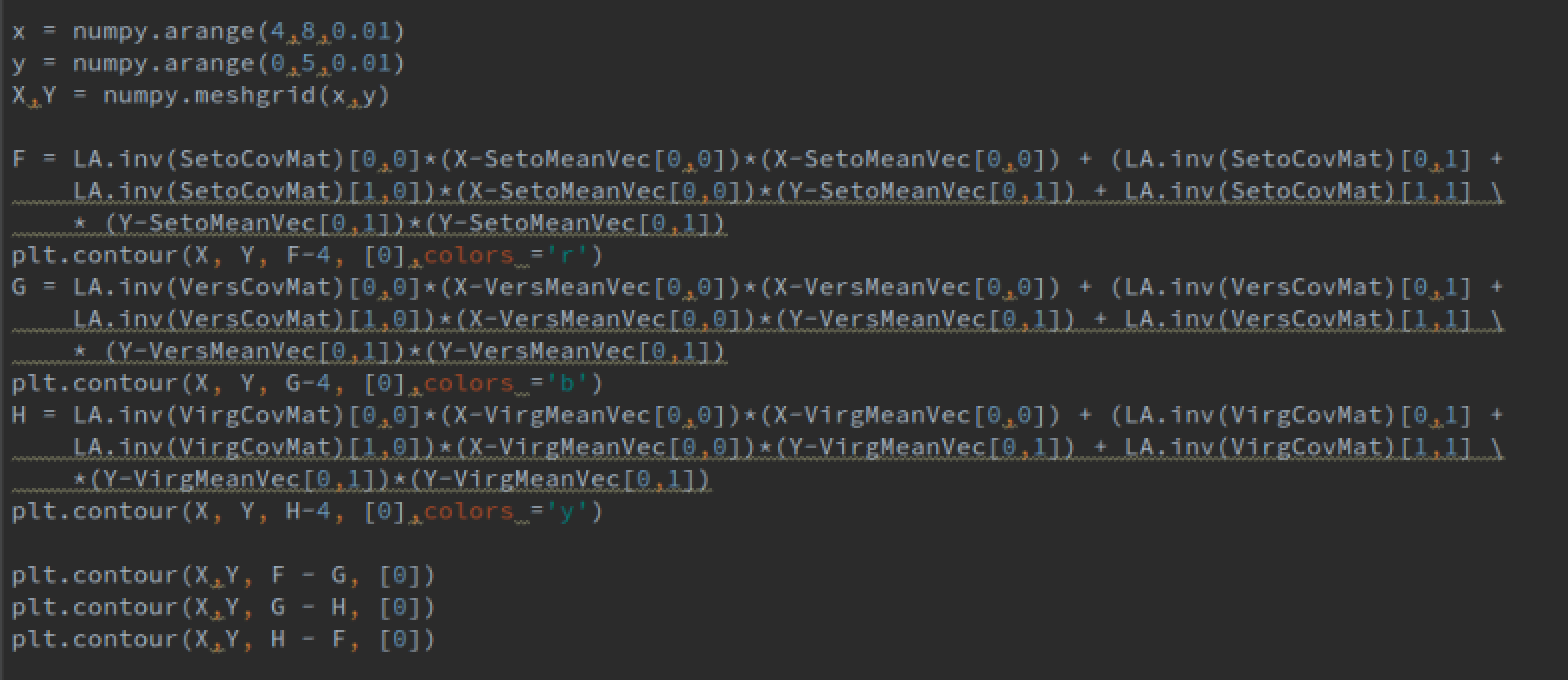
[그림 ] input data 평균 벡터 계산

평균 벡터를 이용하여 공분산 행렬을 구한다.



[그림 ] 공분산 행렬

공식을 이용하여 Mahalanobis 거리를 구하여, 거리가 2인 부분까지 contour로 표현한다.



[그림 ] Mahalanobis 거리를 이용한 범위 계산

구해진 Mahalanobis 거리를 기준으로 3가지 class로 구분할 수 있는 decision boundary 를 가지고 분류한다.

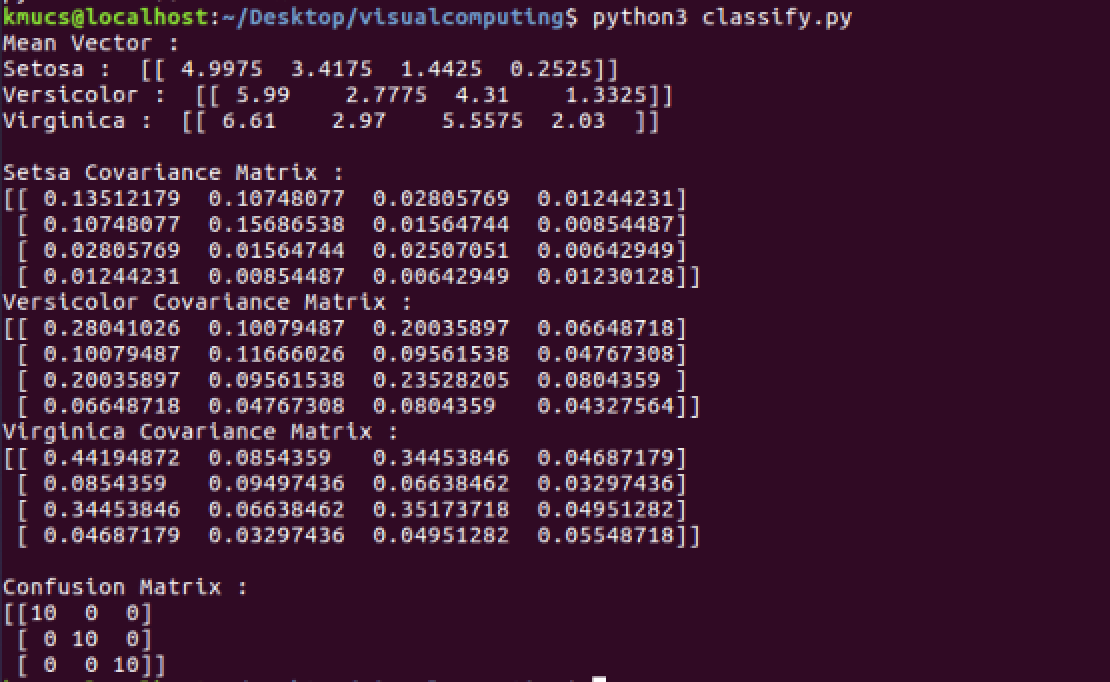
### 활용/개발된 기술

Python에서 행렬 계산과 수식 계산을 용이하게 해주는 “numpy” 모듈을 사용하였다.

그래프의 도식화를 위해 contour와 scatter 등을 사용하는데 이를 제공하는 “pyplot”을 이용하였다.

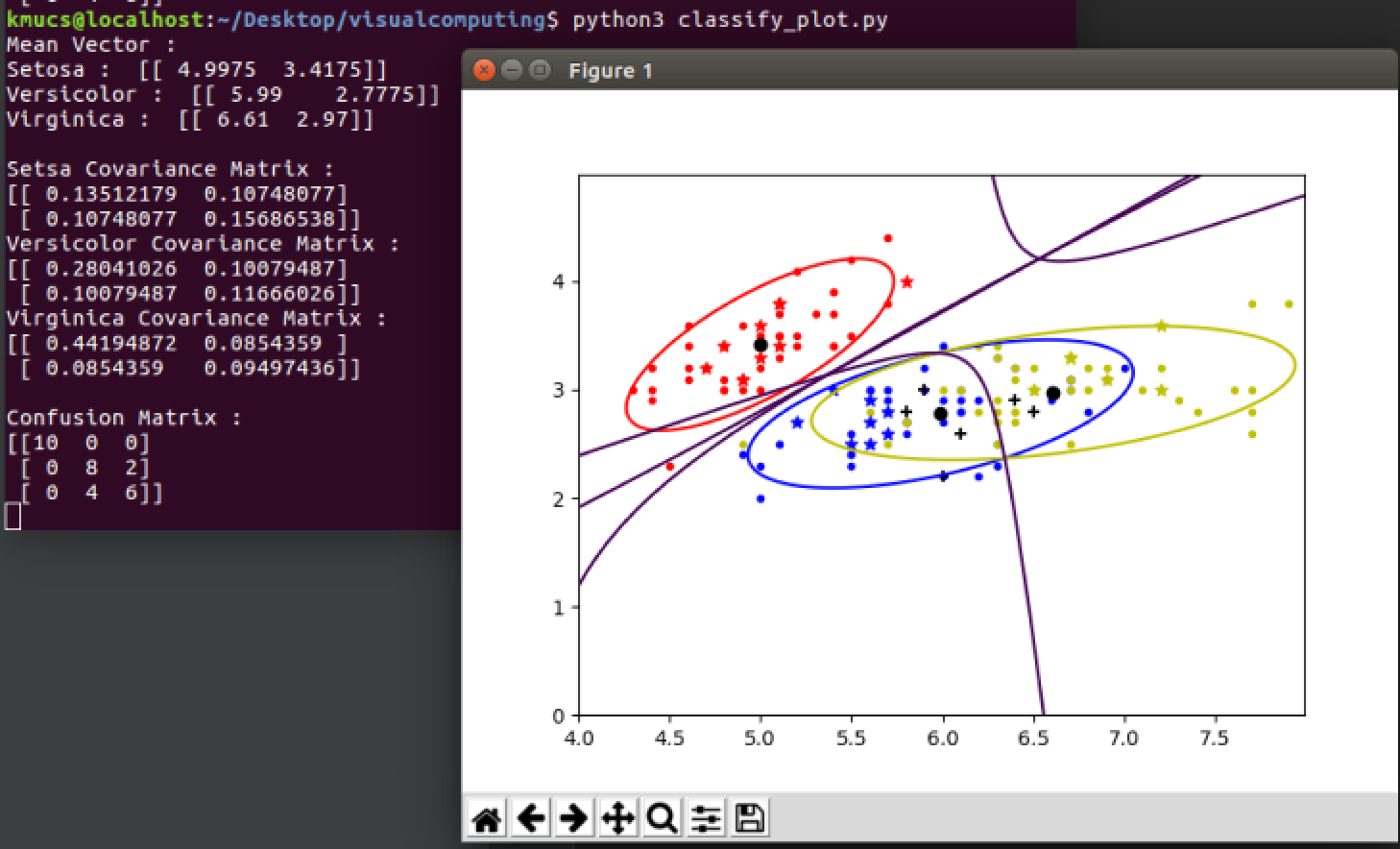
### 결과물 목록

Iris의 4가지 feature를 사용하여 각 class 당 40개의 training set과 10 개의 testing set을 주고 10개의 샘플에 대한 공분산을 구하고 이후 계산을 통한 confusion matrix를 구하는 경우 다음과 같은 결과를 보인다.



[그림 ] 결과 작성

Iris의 2가지 feature를 사용하여 각 class 당 40개의 training set과 10 개의 testing set을 주고 10개의 샘플에 대하여 공분산을 구하고 Mahalanobis 거리를 구하여 분류하는 경우에는 다음과 같은 결과를 보인다.



[그림 ] 결과 작성