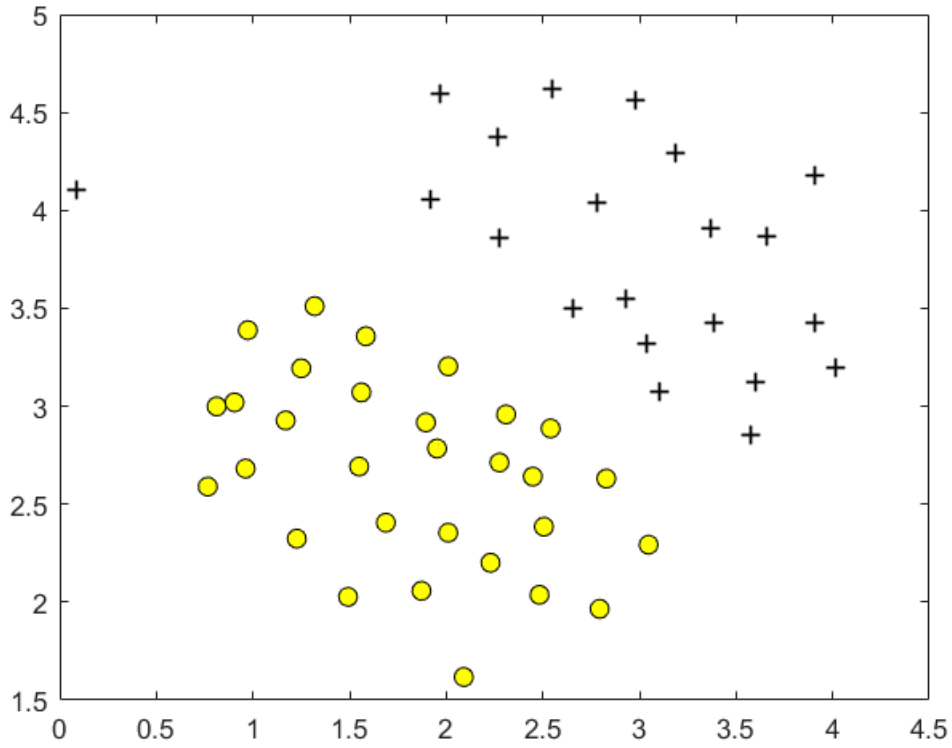
1. Загрузите данные **ex5data1.mat** из файла.

```
load('ex6data1.mat');
```

2. Постройте график для загруженного набора данных: по осям - переменные X1, X2, а точки, принадлежащие различным классам должны быть обозначены различными маркерами.

```
plotData(X, y);
```



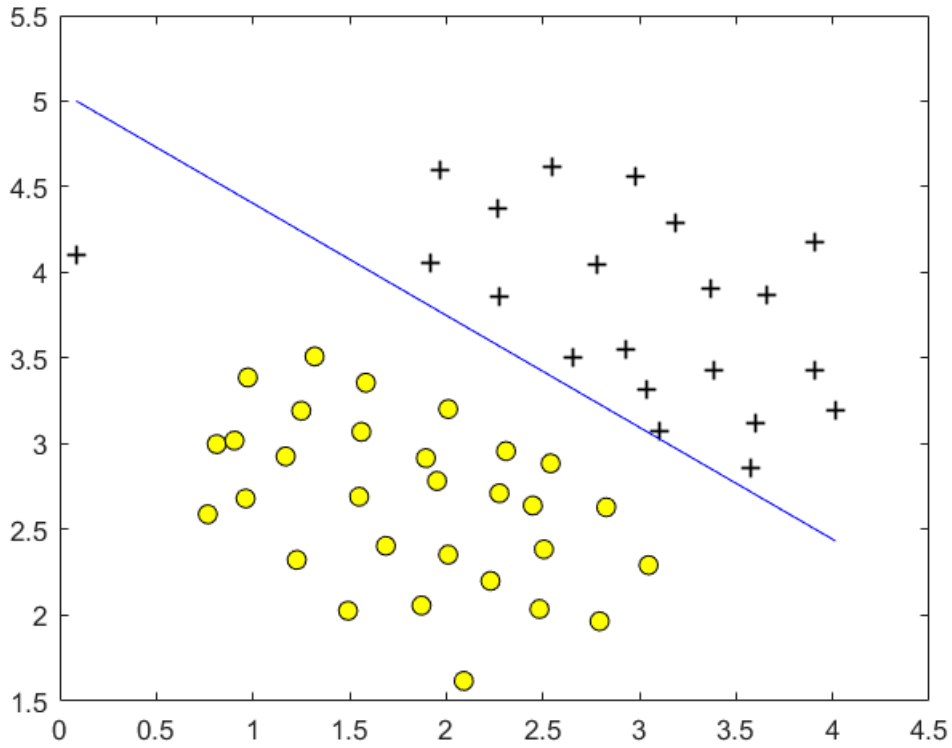
3. Обучите классификатор с помощью библиотечной реализации SVM с линейным ядром на данном наборе.

4. Постройте разделяющую прямую для классификаторов с различными параметрами C = 1, C = 100 (совместно с графиком из пункта 2). Объясните различия в полученных прямых?

```
C = 1;
model = svmTrain(X, y, C, @linearKernel, 1e-3, 20);
```

```
Training ..................................................................... Done!
```

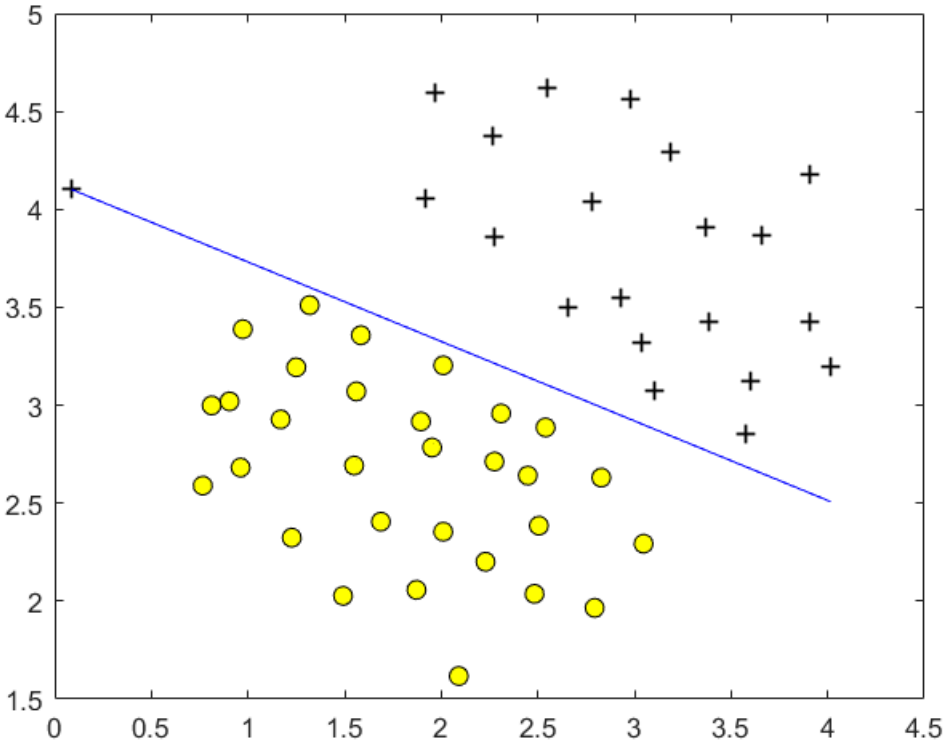
```
visualizeBoundaryLinear(X, y, model);
```




```
C = 100;
model = svmTrain(X, y, C, @linearKernel, 1e-3, 20);
```

```
Training ...........................................................................
.....................................................................................
.....................................................................................
.....................................................................................
.....................................................................................
..................................... Done!
```

```
visualizeBoundaryLinear(X, y, model);
```



При большем C линия разделила все примеры, менее чувствительна к примерам не похожим на общий тренд. При меньшем C линия более похожая на естественную.

5. Реализуйте функцию вычисления Гауссового ядра для алгоритма SVM.

```
function sim = gaussianKernel(x1, x2, sigma)

x1 = x1(:); x2 = x2(:);
sim = 0;
sim = exp(-sum((x1-x2).^2)/(2*sigma^2));
end
```

6. Загрузите данные **ex5data2.mat** из файла.

```
load('ex6data2.mat');
```

7. Обработайте данные с помощью функции Гауссового ядра.

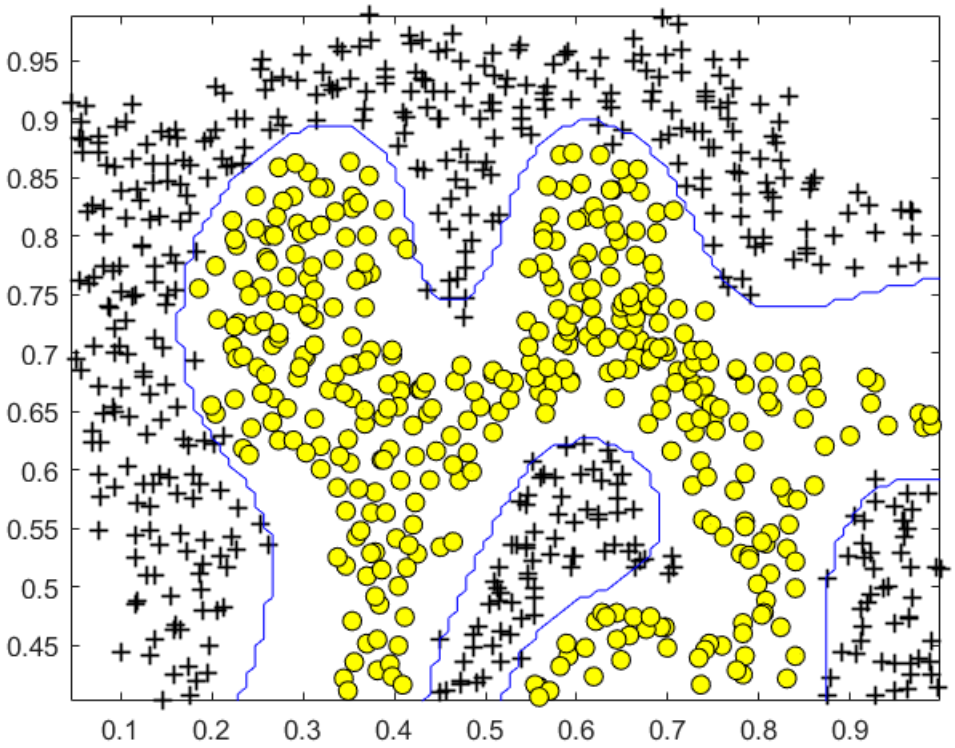
8. Обучите классификатор SVM.

9. Визуализируйте данные вместе с разделяющей кривой (аналогично пункту 4).

```
C = 1; sigma = 0.1;
model= svmTrain(X, y, C, @(x1, x2) gaussianKernel(x1, x2, sigma));
```

```
Training ..........................................................................
..........................................................................
..........................................................................
..........................................................................
..........................................................................
..........................................................................
..........................................................................
..........................................................................
. Done!
```

```
visualizeBoundary(X, y, model);
```



10. Загрузите данные **ex5data3.mat** из файла.

```
load('ex6data3.mat');
```

11. Вычислите параметры классификатора SVM на обучающей выборке, а также подберите параметры C и σ2 на валидационной выборке.

12. Визуализируйте данные вместе с разделяющей кривой (аналогично пункту 4).

```
% Try different SVM Parameters here
[C, sigma] = dataset3Params(X, y, Xval, yval)
```

```
Training ......... Done!


Training ..................................... Done!
```

```
Training ..... Done!

Training ..... Done!

Training ..... Done!

Training ..... Done!

Training ..... Done!

Training ..... Done!

Training ..... Done!

Training ..... Done!

Training ..... Done!

Training ..... Done!

Training ..... Done!

Training ..... Done!

Training ..... Done!

Training ..... Done!

Training ..... Done!

Training .....
. Done!

Training .....
..... Done!

Training ..... Done!

Training ..... Done!

Training ..... Done!

Training ..... Done!
```

```
Training ............................ Done!


Training ......... Done!


Training ................................................................ Done!


Training ........................................................................
.................................................................................
.................................................................................
.................................................................................
..................... Done!


Training ........................................................................
....................... Done!


Training ..................................................... Done!


Training ..................................... Done!


Training ........................................... Done!


Training ..................................................... Done!


Training ......... Done!


Training ........................................................................
.................. Done!


Training ........................................................................
.................................................................................
.................................................................................
.................................................................................
............... Done!


Training ........................................................................
....................... Done!


Training ..................................................... Done!


Training ................................................................ Done!


Training ........................................... Done!


Training ............................................... Done!


Training ......... Done!
```

Training .....
..... Done!

Training .....
.....
.....
.....
.....
.....
Done!

Training .....
.....
.....
..... Done!

Training .....
..... Done!

Training ..... Done!

Training ..... Done!

Training ..... Done!

Training ..... Done!

Training .....
..... Done!

Training .....
.....
.....
.....
.....
.....
.....
.....
.....
.....
..... Done!

Training .....
.....
.....
..... Done!

Training .....
..... Done!

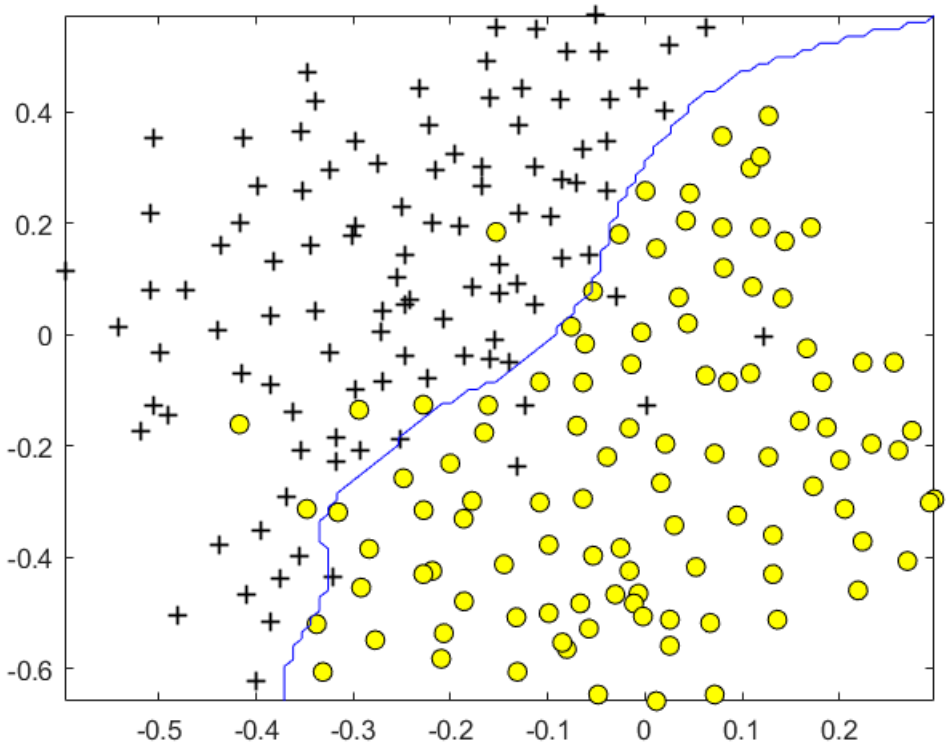
Training ..... Done!

```
Training .....................................................................
.............. Done!


Training ..................................... Done!


Training ......... Done!


Training .....................................................................
................................. Done!


Training .....................................................................
.....................................................................
.....................................................................
.....................................................................
.....................................................................
.....................................................................
.....................................................................
.....................................................................
.....................................................................
.....................................................................
.....................................................................
.....................................................................
.....................................................................
.....................................................................
.....................................................................
.....................................................................
.....................................................................
.....................................................................
........................................ Done!


Training .....................................................................
.....................................................................
.....................................................................
.....................................................................
.....................................................................
........................................................ Done!


Training .....................................................................
....... Done!


Training .....................................................................
........... Done!


Training ............................................... Done!


Training ..................................... Done!
C = 1
sigma = 0.1000
```

```
% Train the SVM
model = svmTrain(X, y, C, @(x1, x2)gaussianKernel(x1, x2, sigma));
```

```
Training ..........................................................................
..........................................................................
..........................................................................
.............................. Done!
```

```
visualizeBoundary(X, y, model);
```



13. Загрузите данные **spamTrain.mat** из файла.

```
load('spamTrain.mat');
```

14. Обучите классификатор SVM.

```
C = 0.1;
model = svmTrain(X, y, C, @linearKernel);
```

```
Training ..........................................................................
..........................................................................
..........................................................................
..................................... Done!
```

```
p = svmPredict(model, X);
fprintf('Training Accuracy: %f\n', mean(double(p == y)) * 100);
```

```
Training Accuracy: 99.825000
```

15. Загрузите данные **spamTest.mat** из файла.

```
load('spamTest.mat');
```

```
Xtest = 1000×1899
     0     0     0     0     0     0     0     0     0     0     0     0 ···
     0     0     0     0     1     0     0     0     0     0     0     0
     0     0     0     0     0     0     0     0     0     0     0     0
     0     0     0     0     0     0     1     0     0     1     0     0     0
     0     0     0     0     0     0     0     0     0     0     0     0     0
     0     0     0     0     0     0     0     0     0     0     0     0     0
     0     0     0     0     0     0     0     0     1     0     0     0     1
     0     0     0     0     0     0     0     0     0     0     0     0     0
     0     0     0     0     0     0     0     0     0     0     0     0     0
     0     0     0     0     0     0     0     0     0     0     0     0     0
     ⋮
```

```
p = svmPredict(model, Xtest);
fprintf('Test Accuracy: %f\n', mean(double(p == ytest)) * 100);
```

```
Test Accuracy: 98.900000
```

```
[C, sigma] = dataset3Params(X, y, Xval, yval)
```

```
Undefined function or variable 'Xval'.
```

17. Реализуйте функцию предобработки текста письма.

18. Загрузите коды слов из словаря **vocab.txt**.

19. Реализуйте функцию замены слов в тексте письма после предобработки на их соответствующие коды.

```
function word_indices = processEmail(email_contents)

vocabList = getVocabList();

word_indices = [];

email_contents = lower(email_contents);

email_contents = regexprep(email_contents, '<[^<>]+>', ' ');

email_contents = regexprep(email_contents, '[0-9]+', 'number');

email_contents = regexprep(email_contents, ...
'(http|https)://[^\s]*', 'httpaddr');

email_contents = regexprep(email_contents, '[^\s]+@[^\s]+', 'emailaddr');

email_contents = regexprep(email_contents, '[$]+', 'dollar');

fprintf('\n==== Processed Email ====\n\n');

l = 0;

while ~isempty(email_contents)
```

```
[str, email_contents] = ...
strtok(email_contents, ...
[' @$/#.-:&*+=[]?!(){},''">_<;%' char(10) char(13)]);
str = regexprep(str, '[^a-zA-Z0-9]', '');
try str = porterStemmer(strtrim(str));
catch str = ''; continue;
end;
if length(str) < 1
continue;
end
idx = strmatch(str, vocabList, 'exact');
if ~isempty(idx)
word_indices = [word_indices; idx];
end
if (l + length(str) + 1) > 78
fprintf('\n');
l = 0;
end
fprintf('%s ', str);
l = l + length(str) + 1;

end
fprintf('\n\n=========================\n');

end
```

20. Реализуйте функцию преобразования текста письма в вектор признаков (в таком же формате как в файлах **spamTrain.mat** и **spamTest.mat**).

```
function x = emailFeatures(word_indices)
n = 1899;

x = zeros(n, 1);

x(word_indices) = 1;

end
```

```
test = emailFeatures(processEmail(readFile('emailSample1.txt')))
```

```
==== Processed Email ====

do you want to make dollarnumb or more per week if you ar a motiv and qualifi
individu i will person demonstr to you a system that will make you dollarnumb
number per week or more thi is not mlm call our number hour pre record number
to get the detail number number number i need peopl who want to make seriou
monei make the call and get the fact invest number minut in yourself now
number number number look forward to your call and i will introduc you to
peopl like yourself who ar current make dollarnumb number plu per week number
number number numberljgvnumb numberleannumberlrmsnumb
numberwxhonumberqiytnumb numberrjuvnumberhqcfnumb numbereidbnumberdmtvlnumb

=========================
test = 1899×1
     0
```

```
0
0
0
0
0
0
0
0
0
0
⋮
```

```
p = svmPredict(model, test)
```

```
p = 1
```