高斯核函数

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

找出最合适的C和sigma（使用CrossValidation集合）

```
C_vec = [0.01, 0.03, 0.1, 0.3, 1, 3, 10, 30]';
sigma_vec = [0.01, 0.03, 0.1, 0.3, 1, 3, 10, 30]';
error = zeros(length(C_vec),length(sigma_vec));

for i = 1:length(C_vec),
for j = 1:length(sigma_vec),
model = svmTrain(X, y, C_vec(i), @(x1, x2) gaussianKernel(x1, x2, sigma_vec(j)));   用C和sigma的组合训练svm
predictions = svmPredict(model, Xval);   得出假设
error(i,j) = mean(double(predictions ~= yval));   计算误差
end
end

[C_op,sigma_op] = find(error == min(min(error)));  找出误差最小值的位置

C = C_vec(C_op);
sigma = sigma_vec(sigma_op);

word_indices = [word_indices ; 18]    在向量word_indices末尾追加一个数字18
```

垃圾邮件预处理

```
str = regexprep(str, '[^a-zA-Z0-9]', '');  清除a-z，A-Z，0-9以外的字符
```

清除词根（这里porterStemmer是一个预先准备好的function，内容很复杂）

```
   try str = porterStemmer(strtrim(str));
   catch str = ''; continue;
   end;

% hdrstart = strfind(email_contents, ([char(10) char(10)]));
% email_contents = email_contents(hdrstart(1):end);
```

替换所有大写字母

```
email_contents = lower(email_contents);
```

清除所有html语言

```
% Looks for any expression that starts with < and ends with > and replace
% and does not have any < or > in the tag it with a space
email_contents = regexprep(email_contents, '<[^<>]+>', ' ');
```

清除数字

```
% Look for one or more characters between 0-9
email_contents = regexprep(email_contents, '[0-9]+', 'number');
```

清除链接地址

```
% Look for strings starting with http:// or https://
email_contents = regexprep(email_contents, ...
                           '(http|https)://[^\s]*', 'httpaddr');
```

正则表达式中\s代表空白字符，[^\s]代表除了空白字符以外的所有字符，*代表重复任意次前面那个字符

清除邮件地址

```
% Look for strings with @ in the middle
email_contents = regexprep(email_contents, '[^\s]+@[^\s]+', 'emailaddr');
```

清除美元符号$

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

strcmp(str1, str2)对比str1和str2，如果相同返回1

```
for i = 1:length(vocabList),
    if(strcmp(str, vocabList{i}) == 1)
        word_indices = [word_indices ; i];
    end
end
```