2.1

2 ①. As we know $g \in \partial f_K(x)$

$f(z) \geq f_K(z) \geq f_K(x) + g^T(z-x)$

and since $f_K(x) = f(x)$

$\therefore f(z) \geq f_K(x) + g^T(z-x) = f(x) + g^T(z-x)$

$\therefore g \in \partial f(x)$

2.2

② . when $1-yw^Tx \leq 0$. $J(w) = 0$. $g = 0$

when $1-yw^Tx > 0$. $J(w) = 1-yw^Tx$. $\nabla = 0 - yx$.

$$\therefore g = \begin{cases} -yx & \text{when } yw^Tx < 1 \\ 0 & \text{otherwise} \end{cases}$$

3.1

3. ⑪ . $\{x \mid w^T x = 0\}$

$l(\hat{y}, y) = \max\{0, -\hat{y}y\}$

$\hat{l}(\hat{y}, y) = \frac{1}{n}\sum_{i=1}^{n} l_i(y, w^T x_i)$

$= \frac{1}{n}\sum_{i=1}^{n} \max\{0, -w^T x_i y\}$.

when y is labeled on the postive side, $w^T x_i > 0$ $y > 0$ $\therefore -w^T x_i y < 0$ $\therefore$ the loss $= 0$.
Likewise for y is labeled on the negative side

$\therefore \hat{l} = \frac{1}{n}\sum_{i=1}^{n} 0 = 0$.

3.2

② $\hat{l}(\hat{y}, y) = \frac{1}{n}\sum_{i=1}^{n} \max\{0, -w^T x_i y\}$.

As proved in 2②.

a subgradient is $g = \begin{cases} -y_i x_i & y_i w^T x_i \leq 0 \\ 0 & y_i w^T x_i > 0 \end{cases}$

$\therefore$ if $y_i w^T x_i \leq 0$

$W_{k+1} = W_k - 1 \times (-y_i x_i) = W_k + y_i x_i$

else:

$W_{k+1} = W_k - 1 \times 0 = W_k$.

$\therefore$ SSGD is the same as Perceptron algorithm.

3.3

(3) As proved before:

$$W = \sum_{i=1}^{n} y_i x_i .$$

$\therefore W$ is a linear combination of all the x_i; in particular, only for those x_i that $y_i W^T x_i \leq 0$ (which means viloates the margin). if a data point is always correctly predicted, it will not be a support vector.

4.1

In [645]:
```
import os
import numpy as np
import pickle
import random
from collections import Counter
from datetime import datetime
import matplotlib.pyplot as plt
%matplotlib inline
import pandas as pd
import math
import copy
```

In [164]:
```
def folder_list(path,label):
    '''
    PARAMETER PATH IS THE PATH OF YOUR LOCAL FOLDER
    '''
    filelist = os.listdir(path)
    review = []
    for infile in filelist:
        if infile.endswith(".txt"):
            file = os.path.join(path,infile)
            r = read_data(file)
            r.append(label)
            review.append(r)
        else:
            continue
    return review



def read_data(file):
    '''
    Read each file into a list of strings.
    Example:
    ["it's", 'a', 'curious', 'thing', "i've", 'found', 'that', 'when', 'willis', 'is', 'not',
'called', 'on',
    ...'to', 'carry', 'the', 'whole', 'movie', "he's", 'much', 'better', 'and', 'so', 'is', '
the', 'movie']
    '''
    f = open(file,encoding='utf-8')
    lines = f.read().split(' ')
    symbols = '${}()[].,:;+-*/&|<>=~" '
    table = str.maketrans("a", "a", symbols)
    words = map(lambda Element: Element.translate(table).strip(), lines)
    words = list(filter(None, words))
    return words

################################################
######## YOUR CODE STARTS FROM HERE. ##########
################################################

def shuffle_data():
    '''
    pos_path is where you save positive review data.
    neg_path is where you save negative review data.
    '''
    pos_path = "/Users/sunevan/Dropbox/Spring 2017/Machine Learning/3/Assignment/data/pos"
    neg_path = "/Users/sunevan/Dropbox/Spring 2017/Machine Learning/3/Assignment/data/neg"

    pos_review = folder_list(pos_path,1)
    neg_review = folder_list(neg_path,-1)

    review = pos_review + neg_review
    random.shuffle(review)

    return review
```

In [714]:
```
# choose the first 1500 for training data
df = shuffle_data()
train_df = df[:1500]
test_df = df[1500:]
```

In [174]:
```
X_train = list()
for i in range(len(train_df)):
    X_train.append(train_df[i][:-1])


X_test = list()
for i in range(len(test_df)):
    X_test.append(test_df[i][:-1])

y_train = list()
for i in range(len(train_df)):
    y_train.append(train_df[i][-1])

y_test = list()
for i in range(len(test_df)):
    y_test.append(test_df[i][-1])
```

5.1

In [182]:
```
def dotProduct(d1, d2):
    """
    @param dict d1: a feature vector represented by a mapping from a feature (string) to a we
ight (float).
    @param dict d2: same as d1
    @return float: the dot product between d1 and d2
    """
    if len(d1) < len(d2):
        return dotProduct(d2, d1)
    else:
        return sum(d1.get(f, 0) * v for f, v in d2.items())
```

In [183]:
```
def increment(d1, scale, d2):
    """
    Implements d1 += scale * d2 for sparse vectors.
    @param dict d1: the feature vector which is mutated.
    @param float scale
    @param dict d2: a feature vector.

    NOTE: This function does not return anything, but rather
    increments d1 in place. We do this because it is much faster to
    change elements of d1 in place than to build a new dictionary and
    return it.
    """
    for f, v in d2.items():
        d1[f] = d1.get(f, 0) + v * scale
```

In [197]:
```
def sparse_count(dataset):
    count_list = list()
    for i in dataset:
        count_list.append(dict(Counter(i)))
    return count_list
```

In [649]:
```
X_train_sparse = sparse_count(X_train)
X_test_sparse = sparse_count(X_test)
```

5.2 Optional

6.1

6. ① $J(w) = \min_{w \in R^n} \frac{\lambda}{2}\|W\|^2 + \frac{1}{m}\sum_{i=1}^{m} \max\{0, 1-y_i w^T x_i\}$

As we already know that for $f = f_1 + \cdots f_m$ / $f_1 \cdots f_m \in R^d \to R$

$\partial f(x) = \partial f_1(x) + \cdots \partial f_m(x)$

$\therefore \partial J(w) = \partial_w(\frac{\lambda}{2}\|w\|^2) + \partial_w(\frac{1}{m}\sum_{i=1}^{m}\max\{0, 1-y_i w^T x_i\}$

for a single data point:

$$\partial J(w) = \underline{\partial w(\frac{\lambda}{2}\|W\|^2)} + \partial_w(\max\{0, 1-y_i w^T x_i\})$$

$$\Downarrow \qquad\qquad \downarrow$$

$$\frac{\lambda}{2}\partial w \qquad \begin{cases} 0 & 1-y_i w^T x_i \leq 0 \\ 0-y_i x_i & 1-y_i w^T x_i > 0 \end{cases}$$

$$\therefore W_{t+1} = \begin{cases} w_t - \eta_t(\lambda w_t) = (1-\eta_t\lambda)w_t & y_i w^T x_i \geq 1 \\ w_t - \eta_t(\lambda w_t - y_i x_i) = (1-\eta_t\lambda)w_t + \eta_t y_i x_i & y_i w^T x_i < 1 \end{cases}$$

6.2

```
In [345]: def pegasos(X,y,Lambda,max_round = 100):
              t = 1
              round_count = 0
              w = dict()
              while round_count < max_round:


                  index_list = list(range(len(X)))
                  #random.shuffle(index_list) # shuffle everytime (Comment out so that I can compare fo
          r 6.4)
                  round_count += 1
                  for i in index_list:
                      t+= 1
                      nt = 1/(t*Lambda)
                      if y[i]*dotProduct(w,X[i]) < 1:
                          increment(w,  -1 * nt * Lambda, w)
                          increment(w,nt*y[i],X[i])
                      else:
                          increment(w,  -1 * nt * Lambda, w)

              return w
```

6.3

In [346]:
```
def pegasos_1(X,y,Lambda,max_round = 100):
    t = 1
    s = 1
    round_count = 0
    W = dict()

    while round_count < max_round:

        index_list = list(range(len(X)))
        #random.shuffle(index_list) # shuffle everytime (Comment out so that I can compare fo
r 6.4)
        round_count += 1
        for i in index_list:
            t+= 1
            nt = 1/(t*Lambda)
            s = (1-nt * Lambda) * s
            if y[i]*s*dotProduct(W,X[i]) < 1:
                increment(W,(1/s)*nt*y[i],X[i])

    increment(W,(s-1),W) #rescale w= W+(s-1)W

    return W
```

6.4

In [720]:
```
startTime = datetime.now()
p_1 = pegasos(X_train_sparse,y_train,0.1,1)
print ("Naive Algroithm runs:",datetime.now() - startTime)
```

```
Naive Algroithm runs: 0:00:11.169105
```

In [721]:
```
startTime = datetime.now()
p_2 = pegasos_1(X_train_sparse,y_train,0.1,1)
print ("Updated Algroithm runs:",datetime.now() - startTime)
```

```
Updated Algroithm runs: 0:00:00.390168
```

In [722]:
```
dotProduct(p_1,p_1)
```

Out[722]:
```
35.327680724503345
```

In [723]:
```
dotProduct(p_2,p_2)
```

Out[723]:
```
35.327680724507125
```

w is almost the same by running both algorithm.

6.5

In [353]:
```
def predict_loss(w,X,y):
    loss = 0
    for i in range(len(X)):
        pred = dotProduct(w,X[i])
        if pred * y[i] < 0:
            loss += 1
    return loss / len(X)
```

6.6

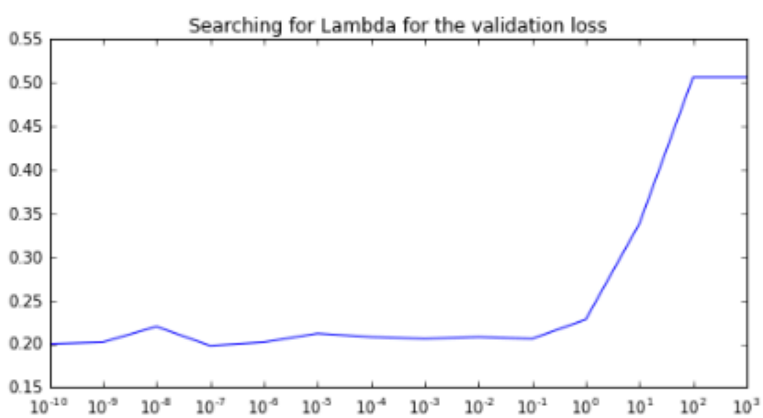
In [438]:
```
# re-define pegasos by adding randomly shuffule data for mutilple round run
def pegasos_2(X,y,Lambda,max_round = 50):
    t = 1
    s = 1
    round_count = 0
    W = dict()

    while round_count < max_round:

        index_list = list(range(len(X)))
        random.shuffle(index_list) # shuffle everytime
        round_count += 1
        for i in index_list:
            t+= 1
            nt = 1/(t*Lambda)
            s = (1-nt * Lambda) * s
            if y[i]*s*dotProduct(W,X[i]) < 1:
                increment(W,(1/s)*nt*y[i],X[i])

    increment(W,(s-1),W) #rescale w= W+(s-1)W

    return W
```

In [446]:
```
w_list = list()
Lambda_list = list()
loss_list = list()

for i in range(-10,4):
    Lambda_list.append(10**i)
    w_inuse = pegasos_2(X_train_sparse,y_train,10**i)
    w_list.append(w_inuse)
    loss = predict_loss(w_inuse,X_test_sparse,y_test)
    loss_list.append(loss)
```

In [671]:
```
best_Lambda = Lambda_list[np.argmin((np.array(loss_list)))]
best_w = w_list[np.argmin((np.array(loss_list)))]
print ("Best Lambda to choose:", best_Lambda)
print ("Minimum loss is:",min(loss_list))
```

```
Best Lambda to choose: 1e-07
Minimum loss is: 0.198
```

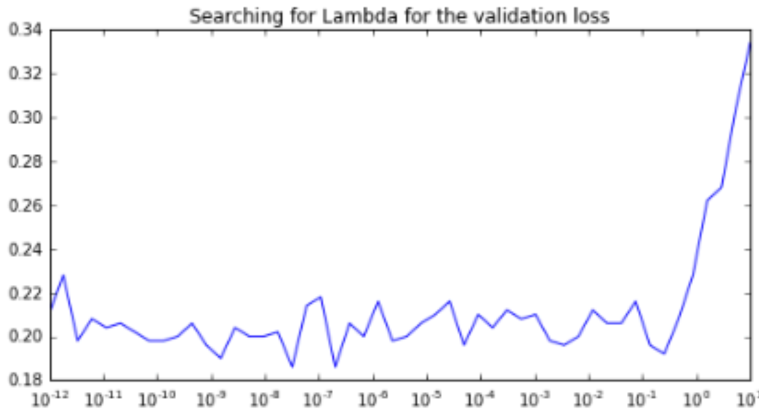
In [447]:
```
fig = plt.figure(figsize = (8,4))
plt.plot(Lambda_list,loss_list)
plt.xscale('log')
plt.title("Searching for Lambda for the validation loss")
plt.show()
```



It seems that when the validation loss increases whenever Lambda is greater than 1. Zoom in Lambda between 1e-12 and 1.

In [448]:
```
w_list_1 = list()
Lambda_list_1 = list()
loss_list_1 = list()

for i in np.linspace(-12,1,num = 50):
    Lambda_list_1.append(10**i)
    w_inuse = pegasos_2(X_train_sparse,y_train,10**i)
    w_list_1.append(w_inuse)
    loss = predict_loss(w_inuse,X_test_sparse,y_test)
    loss_list_1.append(loss)
```

In [449]:
```
fig = plt.figure(figsize = (8,4))
plt.plot(Lambda_list_1,loss_list_1)
plt.xscale('log')
plt.title("Searching for Lambda for the validation loss")
plt.show()
```



In [672]:
```
best_Lambda_1 = Lambda_list_1[np.argmin((np.array(loss_list_1)))]
best_w_1 = w_list_1[np.argmin((np.array(loss_list_1)))]
print ("Best Lambda to choose:", best_Lambda_1)
print ("Minimum loss is:",min(loss_list_1))
```

```
Best Lambda to choose: 3.23745754282e-08
Minimum loss is: 0.186
```

6.7

In [599]:
```
pred_score_list = list()
pred_list = list()
abs_pred_score_list = list()
correction = list()
for i in range(len(X_test_sparse)):
    pred_score_list.append(dotProduct(best_w,X_test_sparse[i]))
    abs_pred_score_list.append(abs(dotProduct(best_w,X_test_sparse[i])))

    if dotProduct(best_w,X_test_sparse[i]) > 0:
        pred_list.append(1)
    else:
        pred_list.append(-1)

    if dotProduct(best_w,X_test_sparse[i])*y_test[i] > 0:
        correction.append(1)
    else:
        correction.append(0)
```

In [600]:
```
df = pd.DataFrame({"pred_score":pred_score_list,"prediction":pred_list,"y_test":y_test,"abs_p
red_score":abs_pred_score_list,\
                   "correction":correction})
df = df.sort(columns="abs_pred_score",ascending=False)

correction_list=list()
max_abs_pred_score = list()
min_abs_pred_score = list()
for i in range(5):
    max_abs_pred_score.append(df.iloc[100*i:100*i+100].max()['abs_pred_score'])
    min_abs_pred_score.append(df.iloc[100*i:100*i+100].min()['abs_pred_score'])
    correction_list.append((df.iloc[100*i:100*i+100].sum()['correction'])/100)
```

```
/Users/sunevan/anaconda/lib/python3.5/site-packages/ipykernel/__main__.py:2: FutureWarning:
sort(columns=....) is deprecated, use sort_values(by=.....)
  from ipykernel import kernelapp as app
```

In [562]:
```
dff = pd.DataFrame({"max_absoulte_pred_score":max_abs_pred_score,"min_absoulte_pred_score":mi
n_abs_pred_score,"correction_pct":correction_list})

dff
```

Out[562]:

	correction_pct	max_absoulte_pred_score	min_absoulte_pred_score
0	0.99	4.612200e+06	1.437323e+06
1	0.93	1.415083e+06	8.982237e+05
2	0.80	8.928698e+05	5.045044e+05
3	0.71	5.028570e+05	2.524581e+05
4	0.64	2.520463e+05	3.294723e+03

So, I group the prediction score into 5 groups by the ranking of the absolute value. As seen in the table, for the prediction score between 4.6 e6 and 1.4 e6, there is only 1 out of 100 texts were predicted to be wrong. The correction % gradually decreses when the score is lower. Also, I guess my prediction score is large due to a small Lambda I choose.

6.8

In [726]:
```
count_1 = 0
total_count =0
for i in X_test_sparse:
    for key in i:
        w = best_w[key] if (key in best_w.keys()) else np.nan
        x = i[key]
        if abs(x*w) <=1.1 and abs(x*w) >= 0.9:
            count_1+=1
        total_count+=1

print (count_1, total_count)
```

```
0 167716
```

It is not surprising to see the result as my Lambda is set to low (which increases the prediction score). I think it is reasonable to skip update when it is equal to 1. However, if the optimal Lambda is close 0 or even larger, then it may not be a good idea.

7.1

By doing 6.7, i am going to use review indexed as 195 and index 337 for the analysis

In [729]:
```
def error_analysis(indexnum):
    print ("This is No.",indexnum,"in the test set")

    if dotProduct(best_w,X_test_sparse[indexnum]) > 0:
        predict_v = 1
    else:
        predict_v = -1

    print ("prediction score:",predict_v)
    print ("y label:",y_test[indexnum])

    dict_xw = dict()
    for key in X_test_sparse[indexnum]:
        w = best_w[key] if (key in best_w.keys()) else np.nan
        x = X_test_sparse[indexnum][key]
        dict_xw[key] = {"abs_xw":abs(x*w),"x":x,"w":w,"xw":(x*w)}
    dict_xw_df = pd.DataFrame.from_dict(dict_xw,'index')
    print(dict_xw_df.sort(columns='abs_xw',ascending=False)[:50],"\n")
    print (' '.join(X_test[indexnum]))
```

In [730]:
```
error_analysis(337)
```

```
This is No. 337 in the test set
prediction score: 1
y label: -1
                    xw             w    x         abs_xw
and       532659.564541  11333.182224   47  532659.564541
the       243196.757376   2133.304889  114  243196.757376
a        -181864.241811  -4133.278223   44  181864.241811
on       -143731.416915 -13066.492447   11  143731.416915
is        118798.416022   3599.952001   33  118798.416022
most      116265.116465  14533.139558    8  116265.116465
at       -107198.570686  -8933.214224   12  107198.570686
by        101998.640018   6799.909335   15  101998.640018
to        -86398.848016  -2399.968000   36   86398.848016
so        -81198.917348 -11599.845335    7   81198.917348
two       -80798.922682 -13466.487114    6   80798.922682
have      -77598.965347 -25866.321782    3   77598.965347
minutes   -73999.013346 -14799.802669    5   73999.013346
this      -65599.125345 -10933.187558    6   65599.125345
then      -64532.472900 -16133.118225    4   64532.472900
you        64399.141345  21466.380448    3   64399.141345
well       61599.178678  20533.059559    3   61599.178678
also       57999.226677  11599.845335    5   57999.226677
in        -57599.232010  -2133.304889   27   57599.232010
lynch     -54265.943121  -4933.267556   11   54265.943121
what       52799.296010   8799.882668    6   52799.296010
several    52799.296010  13199.824002    4   52799.296010
own        51199.317342   8533.219557    6   51199.317342
its        49599.338676   8266.556446    6   49599.338676
original  -48666.017786  -9733.203557    5   48666.017786
enough    -45866.055120 -11466.513780    4   45866.055120
for        43199.424008   3599.952001   12   43199.424008
great      42666.097785  21333.048893    2   42666.097785
any       -40532.792896 -20266.396448    2   40532.792896
may        40532.792896  10133.198224    4   40532.792896
least     -40399.461341 -13466.487114    3   40399.461341
director  -40399.461341 -13466.487114    3   40399.461341
times      39199.477340  13066.492447    3   39199.477340
would     -38399.488007  -9599.872002    4   38399.488007
should    -35999.520007 -17999.760003    2   35999.520007
way        35199.530673  11733.176891    3   35199.530673
work      -32399.568006 -10799.856002    3   32399.568006
those      31999.573339  10666.524446    3   31999.573339
paul      -31599.578672 -10533.192891    3   31599.578672
both       30666.257783  15333.128892    2   30666.257783
only      -30666.257783 -30666.257783    1   30666.257783
better    -29599.605339 -14799.802669    2   29599.605339
he         29199.610672   9733.203557    3   29199.610672
```

he	29199.610672	9733.203557	3	29199.610672
were	28532.952894	14266.476447	2	28532.952894
over	27599.632005	9199.877335	3	27599.632005
why	-27466.300449	-27466.300449	1	27466.300449
one	27332.968894	5466.593779	5	27332.968894
very	26932.974227	13466.487114	2	26932.974227
book	-26666.311116	-6666.577779	4	26666.311116
tv	-26399.648005	-13199.824002	2	26399.648005

the following review encompasses two versions of dune dune the theatrical version 1984 runti me 137 minutes capsule review cut down to just over two hours by nervous studio executives t he theatrical version of dune is a spectacular mess and may be incomprehensible to those unf amiliar with the book the film's visual splendour mystical beauty and impressive action scen es only partly compensate for gaping holes in the narrative dune the extended version 1988 r untime 189 minutes capsule review a bit of a throwtogether assembled by mca tv special proje cts for cable television it was disowned by director david lynch but it's considerably close r to his original vision by virtue of its improved characterisation and clearer storyline qu ality dubs of this version from the outofprint japanese laserdisc release are available from various dealers on the world wide web the review released in 1984 and made on a then mammoth budget of 40 million the film of frank herbert's cult novel dune was eagerly awaited by scif i fans director david lynch blue velvet eraserhead twin peaks was working on his biggest pro duction to date a mammoth undertaking filmed under trying conditions on location in mexico t he screenplay was lynch's own chosen after the script submitted by original author herbert w as rejected dune is set in a universe ruled by powerful families overseen by a successive li ne of emperors the key to cosmic power is the planet arrakis dune a windswept desert planet that's home to giant sandworms and the precious spice melange the spice is the most valuable commodity in the universe it extends the life and expands the consciousness of those who con sume it most importantly it allows the navigators of the spacing guild once human but now hi deously mutated to fold space and navigate their spacecraft across mammoth distances instant aneously enabling interstellar commerce and trade to flourish lynch's film by necessity exci ses parts of the book while retaining the story's two main strands one is the longstanding r ivalry between two families houses atreides and house harkonnen and their battle for lucrati ve mining rights on arrakis the second strand is the emergence of young paul atreides as the reluctant messiah longawaited by the natives of arrakis the fremen the deeply religious frem en want control over their homeworld and young paul may be the fulfilment of their prophecy that a man would come from the outer worlds and lead them to freedom unfortunately this epic story unfolds in a confusing and haphazard manner in the theatrical cut of the film which ru ns 30 to 60 minutes shorter than what lynch originally intended the thinking among universal 's ohsowise money men was that films over two hours in duration were not popular with audien ces at the time and would not do well at the box office with lynch's initial cut running at closer to three or more hours the studio demanded that further cuts be made what a great ide a ! why not trim down an already complex film so as to make it almost incomprehensible ? the most glaring consequence of this oneeyed stupidity is a hopelessly jumpy narrative leaving u s with badly underdeveloped characters thus their personalities are vague their motivations unclear and in the case of paul's father duke leto their demise rather meaningless the end r esult is a distinct chill we can't warm to most of the cast and we don't care much for them and it hardly helps that the voiceover narration is sparse and that the duneesque language a nd terminology sounds like so much gobbledegook to those unfamiliar with the book dune is al so a very serous film the constant selftalk by various characters makes it so serious and se lfabsorbed at times that you may find it hard not to wince with embarrassment the overall im pression is a world full of people so intense that no one is allowed a joke lest the univers e come crashing down around them humour or at least a gentle kind of humour as distinct from the harkonnen's mad sadistic kind is hard to find you may balk at the comparison but as a wr iter lynch could well have done with some lessons from george lucus' star wars trilogy the t heatrical version is still some way from being a complete disaster however it still possesse s enough of lynch's stylistic quirks and enough visual invention to sustain the interest of viewers with a taste for imaginative scifi special effects whiz carlo rambaldi's giant sandw orms are an awesome sight both the production design anthony masters and costume design bob ringwood are striking and original and the magnificent score by toto and brian eno is one of the most underrated soundtracks of the last twenty years with these elements in place and th e benefit of freddie francis' lush cinematography the film is at least a feast for the sense s see it in the widescreen format if you can and despite all the cuts several cast members s till make a strong impression most notably kenneth mcmillan as the supremely nasty baron vla dimir harkonnen sian phillips also registers strongly as the reverend mother gaius helen moh iam leader of the bene gesserit religious order who's secret aim is to manipulate paul's des tiny for its own shadowy ends as paul atreides the young kyle maclachlan starts off somewhat shakily but as his character grows in strength so does his performance and he emerges as a c redible leader of the fremen crusade the conclusion ? any assessment of this film must take into account that frank herbert's original novel is a complex piece of work and presents a t ough challenge for any filmmaker david lynch took a brave stab at it and partly due to force s beyond his control ended up with an officially released version that fails in several key respects dune certainly confused and frustrated a lot of people on its release many chose to stay away altogether as the film's disastrous box office showing attests the extended versio

```
special edition director's cut of the film on home video a clear indication of his dissatisf
action with the version that ended up in the theatres but alas he failed to do so choosing t
o move on to other projects in a way then it is partly lynch's own fault that what appeared
instead was an unauthorised extended version put together in 1988 by mca tv special projects
for airing on cable networks in the usa stung into action lynch successfully petitioned the
director's guild to take his name off the credits and replace it with allen smithee the stan
dard pseudonym for directors who wish to disown their own work he also had the screenwriting
credit changed to the anonymous judas booth certainly looking at the results of mca's handiw
ork there's at least half a dozen instances that for sheer technical sloppiness are good eno
ugh reasons for the director to object but these gripes must be considered in light of the i
mprovements that the extended cut of dune offers in several crucial areas most of the change
s involve the restoration or extension of cut scenes and the addition of extra narration bot
h of which fill many holes in the original version's storyline paul's relationship with his
father and associates is more intimate with moments of humour and warmth lacking previously
the political skulduggery involving the emperor the spacing guild the bene gesserits and the
two warring houses is far better explained paul's initiation into the fremen way of life on
arrakis is also fleshed out considerably and as further background a new prologue has been a
dded featuring narration and painted stills to give us a brief history of the dune universe
as a piece of storytelling then mca tv's version of dune is clearly superior as a piece of e
diting however it is at times surprisingly inept the use of painted stills in the new prolog
ue works well enough but their occasional appearance once the action begins is inappropriate
there's some sloppy cutting too and in a few instances shots even appear out of order and th
e use of repeated footage to fabricate certain scenes eg ships coming and going soldiers com
ing and going is at times clearly outofcontext this is the kind of thing to which lynch obje
cted and rightly so it should also be noted that several questionable scenes and shots from
the theatrical version were deleted to satisfy the censorship demands of u s television but
the most notable omission is a gratuitous piece of nonsense from lynch that wasn't even in h
erbert's book the scene features baron harkonnen killing a beautiful young man in front of h
is slobbering henchmen by pulling out his heart plug its a surreal and disturbing episode th
at's very lynchesque but adds nothing to what we already know the baron is a nasty piece of
work despite its own peculiar flaws then the extended version of dune is a generally superio
r film all up it contains 35 minutes of restored footage and approximately another 15 minute
s of either altered fabricated or newly created sequences unless the idiosyncratic lynch has
a sudden change of heart the alan smithee version remains the closest we'll get to what the
movie should have been on repeated viewings one suspects it is closer than what lynch would
be prepared to admit still as one of this century's great sciencefiction novels some fans an
d perhaps the late herbert himself would argue that dune deserved a better fate in its trans
fer to the screen with rumours circulating of a new six hour miniseries planned by productio
n company new amsterdam entertainment in 1998 it is unlikely that we have heard the last of
the dune saga
```

```
/Users/sunevan/anaconda/lib/python3.5/site-packages/ipykernel/__main__.py:18: FutureWarning:
sort(columns=....) is deprecated, use sort_values(by=.....)
```

Analysis: This is a negative review and is predicted as a positive one. Even though "the" is relatively light weighted, a crazy 114 times apperance still helped a heavy positive weight in this case. In which, I guess I can try to remove stop words like "the", "and". Also, in this praticular case, the author used "most" for 8 times. However, it is used as a superlative in netural tone.

In [731]:
```
error_analysis(195)
```

```
This is No. 195 in the test set
prediction score: -1
y label: 1
                      xw             w   x         abs_xw
and          271996.373383  11333.182224  24  271996.373383
this        -163997.813364 -10933.187558  15  163997.813364
you          107331.902242  21466.380448   5  107331.902242
on          -104531.939574 -13066.492447   8  104531.939574
the          100265.329795   2133.304889  47  100265.329795
a            -82665.564460  -4133.278223  20   82665.564460
two          -53865.948454 -13466.487114   4   53865.948454
plot         -51999.306676 -25999.653338   2   51999.306676
have         -51732.643565 -25866.321782   2   51732.643565
movie        -45732.723564  -6533.246223   7   45732.723564
to           -40799.456008  -2399.968000  17   40799.456008
if           -40266.129785 -20133.064893   2   40266.129785
mind          32399.568006  10799.856002   3   32399.568006
then         -32266.236450 -16133.118225   2   32266.236450
picture       31999.573339  10666.524446   3   31999.573339
other         30666.257783  15333.128892   2   30666.257783
now           29866.268450  14933.134225   2   29866.268450
in           -29866.268450  -2133.304889  14   29866.268450
would        -28799.616005  -9599.872002   3   28799.616005
your          27599.632005   9199.877335   3   27599.632005
director     -26932.974227 -13466.487114   2   26932.974227
or           -26132.984894  -6533.246223   4   26132.984894
much         -25599.658671  -6399.914668   4   25599.658671
during        25599.658671   6399.914668   4   25599.658671
for           25199.664005   3599.952001   7   25199.664005
7             25066.332449   6266.583112   4   25066.332449
than          24799.669338   8266.556446   3   24799.669338
could        -24799.669338 -12399.834669   2   24799.669338
apparently   -24399.674671  -8133.224890   3   24399.674671
depp         -24266.343116  -3466.620445   7   24266.343116
also          23199.690671  11599.845335   2   23199.690671
so           -23199.690671 -11599.845335   2   23199.690671
did          -22799.696004  -7599.898668   3   22799.696004
is            21599.712004   3599.952001   6   21599.712004
those         21333.048893  10666.524446   2   21333.048893
supposed     -19866.401781 -19866.401781   1   19866.401781
he            19466.407115   9733.203557   2   19466.407115
over          18399.754670   9199.877335   2   18399.754670
should       -17999.760003 -17999.760003   1   17999.760003
enjoy         17599.765337   8799.882668   2   17599.765337
be           -17599.765336  -2933.294223   6   17599.765336
too          -17199.770670 -17199.770670   1   17199.770670
own           17066.439114   8533.219557   2   17066.439114
one           16399.781336   5466.593779   3   16399.781336
roles        -15999.786669  -7999.893335   2   15999.786669
without       15999.786669   7999.893335   2   15999.786669
will          15866.455114  15866.455114   1   15866.455114
being         15733.123558   7866.561779   2   15733.123558
role          15599.792003   5199.930668   3   15599.792003
s            -15466.460447  -7733.230224   2   15466.460447
```

```
film adaptation of hunter s thompson's infamous semiautobiographical hallucinogenfueled book
of the same title director terry gilliam of twelve monkeys 810 and brazil 710 fame took over
the helm of this project after fellow director alex cox sid and nancy 7 510 apparently alien
ated everyone associated with the movie according to gilliam plot writer thompson depp heads
down to las vegas with his attorney dr gonzo del toro to cover a motorcycle race during thei
r trip they systematically consume two bags of grass seventyfive pellets of mescaline five s
heets of high powered blotter acid a salt shaker halffull of cocaine a whole galaxy of multi
colored uppers downers screamers laughers a quart of tequila a quart of rum a case of beer a
pint of raw ether and two dozen amyls the movie presents us with the results of that heavy d
rug use critique i have given this movie two separate ratings because i believe that the enj
oyment of this psychedelic picture is highly correlated with the amount of drugs or alcohol
that would be floating around in the viewer's own mind whilst inhaling this cinematic vision
of excess if you are prepared to get high or intoxicated before watching this film i would s
ay that this is one picture that you will thoroughly enjoy on a multitude of colorful levels
if on the other hand you decide to stray from the addition of nefarious elements to your sys
tem i could not imagine you truly appreciating much of this druginduced picture's entire rid
e 5 510 for all those sober dogs note i have not read thompson's book having said that joblo
did engage in an alcoholbased consumatory session before and during the viewing of this film
so his critique of the film should be appreciated on that level this movie relies heavily on
style and peculiar humour rather than substance or plot it moves admirably from one scene to
the next without much basis of their being while presenting us with the two days in the life
of writer hunter s thompson during which he seemed to consume more drugs and alcohol than an
yone could ever imagine it was 1971 and the times were apparently a' changing in the states
johnny depp chews into his role like an overgrown child sucking on a chocolate lollipop duri
ng the filming depp apparently become fast friends with reallife writer thompson and was kno
wn to wander off the set from time to time for the sake of checking out the newest barmaid a
t the local watering hole i thought he did seem to exaggerate his walk a little bit too much
but then again this movie is supposed to be a wild exaggeration of everything and anything s
o who am i to talk the one thing that did blow my mind was the actual physical transformatio
n endured by actor benicio del toro for his role as dr gonzo i couldn't believe that this fa
t samoan lawyer was the same guy who played the slick mumbling criminal in the usual suspect
s 7 510 word on the street is that gained over 40 pounds for this role and i must say that h
is look was deliciously reprehensible plenty of cameos also pepper this kaleidoscopic moving
picture in the form of ellen barkin christina ricci tobey maguire and cameron diaz along wit
h a bunch of others other than that the soundtrack was expectedly eclectic the style was not
as wild as i thought it would be and the ending was certainly not much of a barnburner but t
hen again who really noticed this movie is about visions of bats floating through your head
johnny depp looking goofy and being bald and the cornucopia of drugravaged scenes filling yo
ur own intoxicated system with ideas of anarchy rebellion and the lost american dream and fo
r all those who plan on seeing this movie without the partnership of a mean drink or a might
y doobie i suggest you move further down the aisle buy yourself a ticket to godzilla 610 and
enjoy the visual fabrications manufactured for the unstimulated mind little known facts depp
and del toro snorted plenty of powdered milk instead of cocaine bill murray also portrayed a
thompsonbased character in the film where the buffalo roam johnny depp turned down roles in
the three musketeers speed 7 510 and legends of the fall 7 510 for smaller and quirkier role
s in benny and joon 6 510 and what's eating gilbert grape ? 710 in 1988 depp told rolling st
one magazine that he'd tried every drug by the age of 14 johnny hung out with some of the me
mbers of oasis while filming the uncompleted divine rapture in ireland and later played some
slide guitar on the 1997 album be here now johnny was born in kentucky is a highschool dropo
ut has nicknamed himself mr stench has been engaged to four women until now including actres
s winona ryder whose winona forever tattoo had to be altered to wino forever after their bre
akup currently plays guitar in a band called p and owns the viper room nightclub in l a
```

```
/Users/sunevan/anaconda/lib/python3.5/site-packages/ipykernel/__main__.py:18: FutureWarning:
sort(columns=....) is deprecated, use sort_values(by=.....)
```

Analysis: This article used a few negative words like "apparently", "supposed", which sound negatively. In the orginal context, I think it is just the author's writing style to use meaningless adverb. So, probably a bigram can help solve the problem.

8.1

As analyzed above, I will try remove stop words and bi-gram to improve error rate.

Removing Stop Words

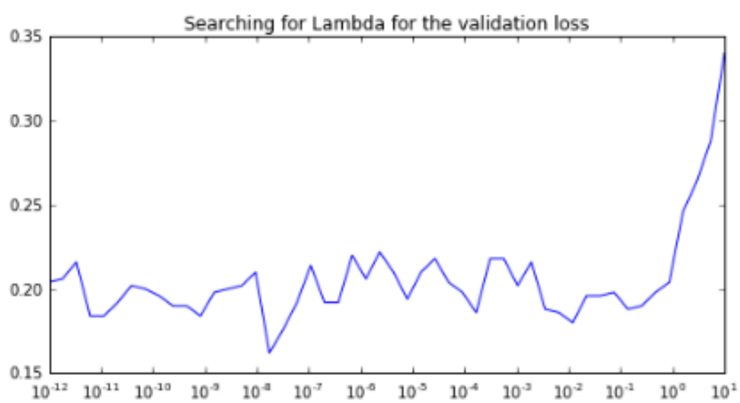
In [773]:
```
from stop_words import get_stop_words
stop_words = get_stop_words('english')
```

In [774]:
```
X_train_sparse_1 = sparse_count(X_train)
X_test_sparse_1 = sparse_count(X_test)
```

In [775]:
```
for i in X_train_sparse_1:
    for j in stop_words:
        try:
            i.pop(j, None)
        except:
            pass
for i in X_test_sparse_1:
    for j in stop_words:
        try:
            i.pop(j, None)
        except:
            pass
```

In [776]:
```
w_list_2 = list()
Lambda_list_2 = list()
loss_list_2 = list()

for i in np.linspace(-12,1,num = 50):
    Lambda_list_2.append(10**i)
    w_inuse = pegasos_2(X_train_sparse_1,y_train,10**i)
    w_list_2.append(w_inuse)
    loss = predict_loss(w_inuse,X_test_sparse_1,y_test)
    loss_list_2.append(loss)
```

In [777]:
```
fig = plt.figure(figsize = (8,4))
plt.plot(Lambda_list_2,loss_list_2)
plt.xscale('log')
plt.title("Searching for Lambda for the validation loss")
plt.show()
```



In [778]:
```
best_Lambda_2 = Lambda_list_2[np.argmin((np.array(loss_list_2)))]
best_w_2 = w_list_2[np.argmin((np.array(loss_list_2)))]
print ("Best Lambda to choose:", best_Lambda_2)
print ("Minimum loss is:",min(loss_list_2))
```

```
Best Lambda to choose: 1.75751062485e-08
Minimum loss is: 0.162
```

In [779]:
```
print ("Minimum loss without removing stop words:",min(loss_list_1))
```

```
Minimum loss without removing stop words: 0.186
```

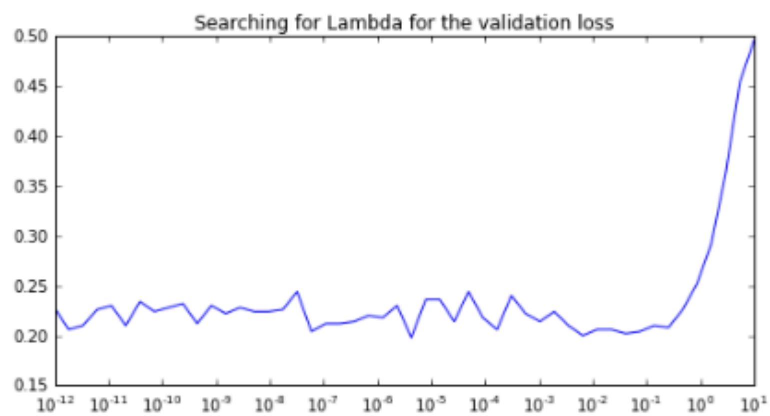
bi-gram

In [783]:
```
def bigram(doc):
    bigram = list()

    for docc in doc:
        doc_dict = dict()
        for i in range(len(docc)):

            if i < len(docc)-1:
                key_inuse =(docc[i],docc[i+1])
                try:
                    doc_dict[key_inuse] = doc_dict[key_inuse]+1
                except:
                    doc_dict[key_inuse] = 1
        bigram.append(doc_dict)
    return (bigram)
```

In [784]:
```
X_train_bigram = bigram(X_train)
X_test_bigram = bigram(X_test)
```

In [785]:
```
w_list_3 = list()
Lambda_list_3 = list()
loss_list_3 = list()

for i in np.linspace(-12,1,num = 50):
    Lambda_list_3.append(10**i)
    w_inuse = pegasos_2(X_train_bigram,y_train,10**i)
    w_list_3.append(w_inuse)
    loss = predict_loss(w_inuse,X_test_bigram,y_test)
    loss_list_3.append(loss)
```

In [786]:
```
fig = plt.figure(figsize = (8,4))
plt.plot(Lambda_list_3,loss_list_3)
plt.xscale('log')
plt.title("Searching for Lambda for the validation loss")
plt.show()
```



In [787]:
```
best_Lambda_3 = Lambda_list_3[np.argmin((np.array(loss_list_3)))]
best_w_3 = w_list_3[np.argmin((np.array(loss_list_3)))]
print ("Best Lambda to choose:", best_Lambda_3)
print ("Minimum loss is:",min(loss_list_3))
```

```
Best Lambda to choose: 4.29193426013e-06
Minimum loss is: 0.198
```

So, after removing the stop words, the best error rate is 0.162. The original standard error is 0.01740. The new standard error is 0.0164. It seems that removing stop words doesn't signficantly improve the rate. And, when using bi-gram instead of single word, the error rate is even higher.

8.2 – Optional