In [1]:
```
# 导入库
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.datasets import load_breast_cancer
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import GridSearchCV
from sklearn.model_selection import cross_val_score
```

In [2]:
```
# 导入数据
breast = load_breast_cancer()
breast.data.shape
```

Out[2]:
```
(569, 30)
```

In [3]:
```
breast.target.shape
```

Out[3]:
```
(569,)
```

In [4]:
```
# 简单建模
rfc = RandomForestClassifier(n_estimators=100, random_state=0)
score_pre = cross_val_score(rfc, breast.data, breast.target, cv=10).mean()
score_pre
```

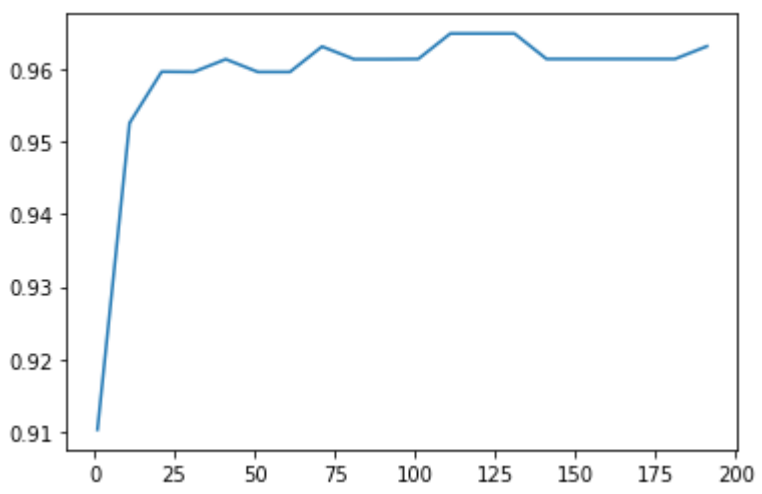
Out[4]:
```
0.9649122807017545
```

In [5]:
```
# 先调 n_estimators
# 学习曲线
scorel = []
for i in range(0,200,10):
  rfc = RandomForestClassifier(n_estimators = i+1
                               ,n_jobs=-1
                               ,random_state=0)
  score = cross_val_score(rfc, breast.data, breast.target, cv=10).mean()
  scorel.append(score)

print(max(scorel), scorel.index(max(scorel))*10+1)
```

```
0.9649122807017545 111
```

In [6]:
```
plt.plot(range(1,201,10), scorel)
plt.show()
```

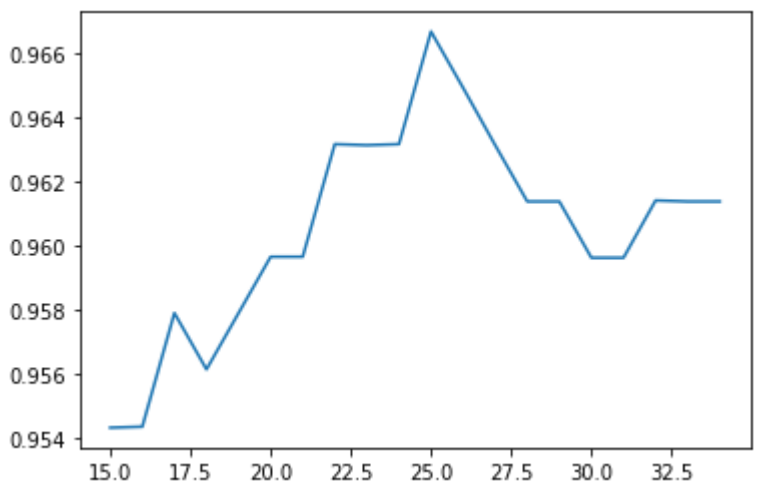


In [7]:
```
# 进一步细化学习曲线（15～35）
scorel = []
for i in range(15,35):
  rfc = RandomForestClassifier(n_estimators = i
                               ,n_jobs=-1
                               ,random_state=0)
  score = cross_val_score(rfc, breast.data, breast.target, cv=10).mean()
  scorel.append(score)

print(max(scorel), scorel.index(max(scorel))+15)
```

```
0.9666666666666666 25
```

In [8]:
```
plt.plot(range(15,35), scorel)
plt.show()
```



In [10]:
```
# 调整 max_depth
param_grid = {'max_depth': np.arange(1,10,1)}

rfc = RandomForestClassifier(n_estimators=25, random_state=0)
GS = GridSearchCV(rfc, param_grid, cv=10)
GS.fit(breast.data, breast.target)

print(GS.best_params_, GS.best_score_)
```

```
{'max_depth': 7} 0.9649122807017543
```

In [11]:
```
# 调整 max_features
param_grid = {'max_features': np.arange(5,30,1)}  # 默认 sqrt(n_features) = sqrt(30)

rfc = RandomForestClassifier(n_estimators=25, random_state=0)
GS = GridSearchCV(rfc, param_grid, cv=10)
GS.fit(breast.data, breast.target)

print(GS.best_params_, GS.best_score_)
```

```
{'max_features': 5} 0.9666666666666666
```

In [13]:
```
# 调整 min_samples_leaf
param_grid = {'min_samples_leaf': np.arange(1,11,1)}

rfc = RandomForestClassifier(n_estimators=25, random_state=0)
GS = GridSearchCV(rfc, param_grid, cv=10)
GS.fit(breast.data, breast.target)

print(GS.best_params_, GS.best_score_)
```

```
{'min_samples_leaf': 1} 0.9666666666666666
```

In [14]:
```
# 调整 min_samples_split
param_grid = {'min_samples_split': np.arange(2,21,1)}

rfc = RandomForestClassifier(n_estimators=25, random_state=0)
GS = GridSearchCV(rfc, param_grid, cv=10)
GS.fit(breast.data, breast.target)

print(GS.best_params_, GS.best_score_)
```

```
{'min_samples_split': 2} 0.9666666666666666
```

In [15]:
```
# 最后调整 criterion
param_grid = {'criterion': ['gini', 'entropy']}

rfc = RandomForestClassifier(n_estimators=25, random_state=0)
GS = GridSearchCV(rfc, param_grid, cv=10)
GS.fit(breast.data, breast.target)

print(GS.best_params_, GS.best_score_)
```

```
{'criterion': 'gini'} 0.9666666666666666
```

In [16]:
```
# 最佳模型
rfc = RandomForestClassifier(n_estimators=25, random_state=0)
score = cross_val_score(rfc, breast.data, breast.target, cv=10).mean()

score - score_pre
```

Out[16]:
```
0.0017543859649120641
```