In [6]:
```
# 导入操作系统库
import os
# 更改工作目录
os.chdir(r"D:\softwares\applied statistics\pythoncodelearning\chap1\sourcecode")
# 导入基础计算库
import numpy as np
# 导入绘图库
import matplotlib.pyplot as plt
# 导入回归器
from sklearn.linear_model import RANSACRegressor, LinearRegression
# 导入数据集生成工具
from sklearn.datasets import make_regression
# 导入绘图库中的字体管理包
from matplotlib import font_manager
# 实现中文字符正常显示
font = font_manager.FontProperties(fname=r"C:\Windows\Fonts\SimKai.ttf")
# 使用seaborn风格绘图
plt.style.use("seaborn-v0_8")
# 设置样本量和离群值的样本量
n_samples = 1000
n_outliers = 50
# 生成数据集
X, y, coef = make_regression(
    n_samples=n_samples,
    n_features=1,
    n_informative=1,
    noise=10,
    coef=True,
    random_state=0,
)
# 添加离群值
# Add outlier data
np.random.seed(0)
X[:n_outliers] = 3 + 0.5 * np.random.normal(size=(n_outliers, 1))
y[:n_outliers] = -3 + 10 * np.random.normal(size=n_outliers)
# 构建线性模型
lr = LinearRegression()
# 模型拟合
lr.fit(X, y)
# 构建稳健回归
ransac = RANSACRegressor()
# 模型拟合
ransac.fit(X, y)
inlier_mask = ransac.inlier_mask_
outlier_mask = np.logical_not(inlier_mask)
# 预测
line_X = np.arange(X.min(), X.max())[:, np.newaxis]
# 线性模型预测
line_y = lr.predict(line_X)
# 稳健回归预测
line_y_ransac = ransac.predict(line_X)
print("Estimated coefficients (true, linear regression, RANSAC):")
print(coef, lr.coef_, ransac.estimator_.coef_,sep="\n")
fig, ax = plt.subplots(figsize=(6,6), tight_layout=True)
# 散点图
ax.scatter(
    X[inlier_mask], y[inlier_mask],
    color="yellowgreen", marker=".", label="Inliers"
```

```
)
ax.scatter(
    X[outlier_mask], y[outlier_mask],
    color="gold", marker=".", label="Outliers"
)
ax.plot(line_X, line_y, color="navy", linewidth=2, label="Linear regressor")
ax.plot(
    line_X,
    line_y_ransac,
    color="cornflowerblue",
    linewidth=2,
    label="RANSAC regressor",
)
ax.legend(loc="lower right")
ax.set_xlabel("Input")
ax.set_ylabel("Response")
plt.show()
fig.savefig("../codeimage/code27.pdf")
```

```
Estimated coefficients (true, linear regression, RANSAC):
82.1903908407869
[54.17236387]
[82.08533159]
```