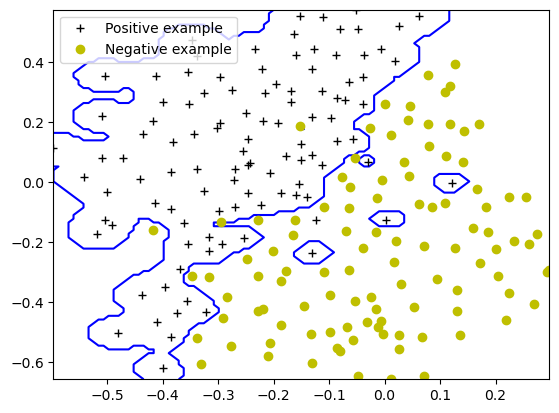
**Проблема: переобучил SVM модель**

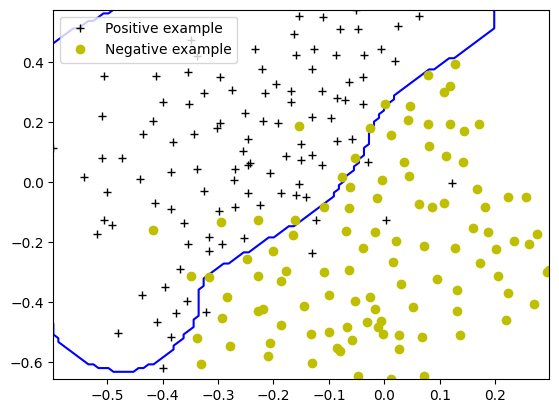
| **def** **dataset3Params**(X, y, Xval, yval):  '''  DATASET3PARAMS возвращает искомые параметры C и sigma для третьей части  упражнения, в котором требуется определить оптимальные значения (C, sigma)  для эффективного использования SVM с некоторой радиальной базисной функцией  (например, c гауссовским ядром)   Следует запрограммировать функцию, используя метод перекрестной проверки (кросс-валидация).   Указание: Необходимо также рассчитать ошибку для набора данных, выбранных для проверки.  Ошибка определяет долю примеров для перекрестной проверки, классифицированных  неправильно.  '''  values = [0.01, 0.03, 0.1, 0.3, 1, 10, 30] **# Набор возможных значений гиперпараметров**  predictions = dict()  **for** C **in** values:  **for** sigma **in** values:  model = svmTrain(X, y, C, 'gaussian\_rbf', sigma, tol = 1e-5, max\_passes = -1)  prediction = model.predict(gaussianKernelGramMatrix(Xval, X, sigma))  predictions[(C, sigma)] = np.mean((prediction != yval))  C, sigma = min(predictions, key=predictions.get)    **return** C, sigma |
| --- |

****

И precision и recall судя по графику имеют значения стремящиеся чуть ли не к 100%  
Модель не имеет обобщающий характер. Точно классифицирует то, что не должна.  
**Результат перекрёстной проверки:**  
Best parameters are C=1.00, sigma=0.03  
  
  
  
**Один из способов решения:**Найти другие гиперпараметры модели.  
в самой ф-ии **dataset3Params** просто убрал это значение 0.03.  
теперь ищу комбинации пар возможных из

| values = [0.01, 0.1, 0.3, 1, 10, 30] |
| --- |

**Результат перекрёстной проверки:**  
Best parameters are C=0.10, sigma=0.10

Обучаем и строим модель с такими параметрами  


**Результат, очевидно, лучше.**