1-We define the multivariate polynomial

$$S = ((\mathbf{x}_i, y_i))_{i=1}^m$$

$$p_S(\mathbf{x}) = -\prod_{i \in [m]: y_i = 1} \|\mathbf{x} - \mathbf{x}_i\|^2$$

Therefore for every y=1 we have p(x)=0 and for other x p(x) Is negative, It follows that learning the class of all thresholded polynomials using the ERM rule may lead to overfitting.

2-

$$\begin{aligned}
\mathop{\mathbb{E}}_{S|_x \sim \mathcal{D}^m}[L_S(h)] &= \mathop{\mathbb{E}}_{S|_x \sim \mathcal{D}^m}\left[\frac{1}{m}\sum_{i=1}^m \mathbb{1}_{[h(x_i) \neq f(x_i)]}\right] \\
&= \frac{1}{m}\sum_{i=1}^m \mathop{\mathbb{E}}_{x_i \sim \mathcal{D}}[\mathbb{1}_{[h(x_i) \neq f(x_i)]}] \\
&= \frac{1}{m}\sum_{i=1}^m \mathop{\mathbb{P}}_{x_i \sim \mathcal{D}}[h(x_i) \neq f(x_i)] \\
&= \frac{1}{m} \cdot m \cdot L_{(\mathcal{D},f)}(h) \\
&= L_{(\mathcal{D},f)}(h) \ .
\end{aligned}$$

3-By definition all the positive instances in the training set are labeled positive by A algorithm.

Because of the realizability and the fact that the tightest rectangle enclosing all positive examples is returned, A can label all the negative instances correctly. Therefore A is an ERM.

We consider the distribution D over X and by hint we define R*.

S is the training set, f is the hypothesis associated with R* and R(S) is the rectangle returned by the algorithm A .The definition of the algorithm A implies that R(S) ⊆ R∗ for every S. Thus,

$$L_{(\mathcal{D},f)}(R(S)) = \mathcal{D}(R^\star \setminus R(S))$$

Fix some € ∈ (0,1). Define R1,R2,R3 and R4 as in the hint. For each i ∈ [4], define the event

$$F_i = \{S|_x : S|_x \cap R_i = \emptyset\}$$

Applying the union bound, we have

$$\mathcal{D}^m(\{S : L_{(\mathcal{D},f)}(A(S)) > \epsilon\}) \le \mathcal{D}^m\left(\bigcup_{i=1}^{4} F_i\right) \le \sum_{i=1}^{4} \mathcal{D}^m(F_i)$$

Thus, it suffices to ensure that Dm(Fi) ≤ δ/4 for every i. Fix some i ∈ [4]. Then, the probability that a sample is in Fi is the probability that all of the instances don't fall in Ri, which is exactly (1−€/4)m. Therefore,

$$\mathcal{D}^m(F_i) = (1 - \epsilon/4)^m \le \exp(-m\epsilon/4)$$

and

$$\mathcal{D}^m(\{S : L_{(\mathcal{D},f)}(A(S)) > \epsilon]\}) \le 4\exp(-m\epsilon/4)$$

Plugging in the assumption on m, we conclude our proof.

The hypothesis class of axis aligned rectangles in Rd is defined as follows. Given real numbers a1 ≤ b1,a2 ≤ b2,...,ad ≤ bd, define the classifier h(a1,b1,...,ad,bd) by

$$h_{(a_1,b_1,\ldots,a_d,b_d)}(x_1,\ldots,x_d) = \begin{cases} 1 & \text{if } \forall i \in [d],\ a_i \leq x_i \leq b_i \\ 0 & \text{otherwise} \end{cases}$$

The class of all axis-aligned rectangles in R^d is defined as

$$\mathcal{H}^d_{rec} = \{h_{(a_1,b_1,\ldots,a_d,b_d)} : \forall i \in [d],\ a_i \leq b_i,\}$$

It can be seen that the same algorithm proposed above is an ERM for this case as well. The sample complexity is analyzed similarly. The only difference is that instead of 4 strips, we have 2d strips (2 strips for each dimension). Thus, it suffices to draw a training set of size [2d log(2d/δ)] /€.

For each dimension, the algorithm has to find the minimal and the maximal values among the positive instances in the training sequence. Therefore, its runtime is O(md). Since we have shown that the required value of m is at most [2d log(2d/δ)] /€., it follows that the runtime of the algorithm is indeed polynomial in d,1/€, and log(1/δ).