1. Let X_i be the indicator variable denoting whether $r_A(s_i) \in [t-\sqrt{n}, t+\sqrt{n}]$. Observe that for any given $t \in [1, n]$,

$$\Pr[X_i = 0] \leq \left(1 - \frac{1}{\sqrt{n}}\right) \leq e^{-1/\sqrt{n}}.$$

Because s_i's are picked independently,

$$\Pr\left[\bigcap_{i=1}^{k}(X_i = 0)\right] \leq e^{-k/\sqrt{n}} \leq e^{-c}$$

if $k \geq cn$ for some constant $c > 0$. Consequently,

$$\Pr[\text{some } s_i \text{ in } S \text{ has rank } r_A(s_i) \in [t-\sqrt{n}, t+\sqrt{n}]] \geq 1 - 1/e^c.$$

2. Let G be any simple graph of average degree $n/3$. Suppose that t nodes in G have degree $\geq n/4$. Then we get

$$\begin{aligned}\sum_{x \in G} \deg(x) &\leq \sum_{x \in G,\ deg(x) \geq n/4} n + \sum_{x \in G,\ deg(x) < n/4} n/4 \\ &= tn + (n-t)n/4 \\ &= n\left(\frac{n+3t}{4}\right)\end{aligned}$$

Because G has the average degree $n/3$, we have

$$\frac{n}{3} \leq \frac{n+3t}{4}$$

or equivalently $t \geq n/9 = \Omega(n)$ as desired.

3. Let X_k be the indicator variable denoting whether element k is included in S_n. Let $X = \sum_{k=1}^{n} X_k$. By calculus, we get

$$\mu = \mathbb{E}[X] = \Theta(\log n).$$

We are done by Chernoff bound

$$\Pr[|X - \mathbb{E}[X]| \geq \varepsilon \mathbb{E}[X]] \leq e^{-\Omega(\varepsilon^2 \mu)} \leq 1/n^{\Omega(\varepsilon^2)}.$$

4. Let $X_{i,j,k}$ be the indicator variable denoting whether nodes i, j, k in G form an triangle. Clearly, $X_{i,j,k} = 0$ is a monotone decreasing graph property for each $i, j, k \in \binom{[n]}{3}$ and

$$\Pr[X_{i,j,k} = 0] = 1 - \frac{c^3}{n^3}.$$

Hence, we get

$$\begin{aligned}
\Pr[G \text{ is triangle-free}] &= \Pr\left[\bigcap_{i,j,k\in\binom{[n]}{3}} X_{i,j,k} = 0\right] \\
&\geq \prod_{i,j,k\in\binom{[n]}{3}} \Pr[X_{i,j,k} = 0] \\
&\geq \left(1 - \frac{c^3}{n^3}\right)^{n^3} \\
&\geq e^{-2c^3} && (\text{if } n^3 \geq 2c^3)
\end{aligned}$$

We are done because c is a constant and n is sufficiently large.

5. Let S be an n-point set. Let R_1 (resp. R_2) be any subset of S so that all points in R_1 (resp. R_2) are covered by an axis-parallel square, but none of the points in $S \setminus R_1$ (resp. $S \setminus R_2$) is covered by the square. Observe that there are $O(n^4)$ such R_1's and R_2's.

For $k = 2$, our algorithm outputs two squares. Let R_1 be the point set covered by the first square, and let R_2 be the point set covered by the second square. In what follows, we will say our algorithm outputs $R_1 \cup R_2$ for simplicity. The points that are not covered by the two squares form the complement set of $R_1 \cup R_2$, i.e. $S \setminus (R_1 \cup R_2)$. There are $O(n^8)$ such complement sets. Our algorithm needs to ensure that, if the complement set of $R_1 \cup R_2$ is large, then it is unlikely to output R_1 and R_2. Here is how.

Let $S = \{p_i : i \in [n]\}$;
$X \leftarrow \emptyset$;
for $i \leftarrow 1$ to $\lceil \frac{9}{\varepsilon} \log n \rceil$ **do**
    Let j be an uniformly random number in $[1, |S|]$;
    $X \leftarrow X \cup \{p_j\}$;
end
Find the optimum $\ell(2)$ to cover X, using $O((1/\varepsilon) \log n)$ space;

Algorithm 1: Pseudocode.

Note that if $S \setminus (R_1 \cup R_2)$ has more than εn points, then X is likely to have an non-empty intersection with $S \setminus (R_1 \cup R_2)$ and Algorithm 1 is unlikely to output $R_1 \cup R_2$.

Here we bound the probability for a single large complement set and get:

$$\Pr\left[(S \setminus (R_1 \cup R_2)) \cap X = \emptyset\right] < (1-\varepsilon)^{\lceil (9/\varepsilon)\log n \rceil} \le e^{-9\log n} = \frac{1}{n^9}.$$

Hence, by Union Bound, Algorithm 1 outputs $R_1 \cup R_2$ whose complement set $S \setminus (R_1 \cup R_2)$ has size no more than εn with probability at least $1 - 1/n$.

To find $\ell(2)$ for X, it needs $O(|X| \log |X|)$ runtime by binary search among all possible $\ell(2)$ and verify each guess using $O(|X|)$ time. The runtime is therefore

$$O((1/\varepsilon)(\log n)(\log(1/\varepsilon) + \log\log n)) = O((1/\varepsilon)\log^2 n).$$

The last equality holds because $\varepsilon \ge 1/n$.

For $k = 3$, we set Line 3 in Algorithm 1 to be a for-loop of $\lceil \frac{13}{\varepsilon} \log n \rceil$ iterations. The remaining part in the proof is similar.