Learning with Errors (LWE) - [Regev 2005]:

Parameters: n (security parameter)

$\alpha = 1/poly(n)$ noise parameter

$q \geq poly(n)$, sometimes even $q = exp(n)$

* For any fixed $\vec{s} \in \mathbb{Z}_q^n$, denote the distribution

$$LWE_{\vec{s}} = \{ (\vec{a}, b) \mid \vec{a} \in_R \mathbb{Z}_q^n, \; e \leftarrow \Phi_{\alpha q}, \; b = \langle \vec{a}, \vec{s} \rangle + e \bmod q \}$$

Where $\Phi_{\alpha q}$ is a distribution over $\mathbb{R}$ with variance $\sim (\alpha q)^2$.
(We can think of it as continuous Gaussian with variance $(\alpha q)^2$ or the same rounded to nearest integer, for now it will be just an arbitrary distribution.)

* Search-LWE: S-LWE$[n, q, \alpha]$: Given oracle access to $LWE_{\vec{s}}$ for a uniformly chosen $\vec{s} \in_R \mathbb{Z}_q^n$, recover $\vec{s}$ (in poly(n) time).

* Decision-LWE: D-LWE$[n, q, \alpha]$: Given an oracle $\mathcal{O}$, decide if $\mathcal{O} = LWE_{\vec{s}}$ for a uniformly chosen $\vec{s} \in_R \mathbb{Z}_q^n$ or if $\mathcal{O} =$ Uniform in $\mathbb{Z}_q^{n+1}$

Thm 1: Given a distinguisher D for D-LWE with advantage $\varepsilon > 1/poly(n)$, can construct another distinguisher D' such that for <u>every</u> $\vec{s}$ D' distinguishes between $LWE_{\vec{s}}$ and Uniform with advantage $1 - 2^{-n}$.

Proof:

We use random-self-reduction + amplification. Consider the following transformation, for some $\vec{r} \in \mathbb{Z}_q^n$ ($\vec{r}$ fixed)

$$\psi_{\vec{r}}(\vec{a}, b) = (\vec{a}, b + \langle \vec{a}, \vec{r} \rangle \bmod q)$$

Note: If $(\vec{a}, b) \in_R$ Uniform then $\psi_{\vec{r}}(\vec{a}, b) \in_R$ Uniform
If $(\vec{a}, b) \in_R LWE_{\vec{s}}$ then $\psi_{\vec{r}}(a, b) \in_R LWE_{\vec{s}+\vec{r}}$

The distinguisher $D^{\mathcal{O}}$ (with access to oracle $\mathcal{O} = LWE_{\vec{s}}$ or Uniform):

0. Using sampling, find threshold value τ such that

$$\Pr_{\vec{s},\$}[D[LWE_{\vec{s}}]=1] > \tau + \varepsilon/4 \quad \text{and} \quad \Pr_{\$}[D[\text{Uniform}]=1] < \tau - \varepsilon/4$$

1. Repeat $N = poly(n, 1/\varepsilon)$ times:
   (i) Choose a random $\vec{r}_i \in_R \mathbb{Z}_q^n$
   (ii) Run the distinguisher D, for each $\mathcal{O}$-query of D sample $(\vec{a},b) \leftarrow \mathcal{O}$ from your own oracle and return $\Psi_{\vec{r}}(\vec{a},b)$
   (iii) Record the answer of D as a vote $V_i \in \{0,1\}$

2. If more than $\tau \cdot N$ of the V_i's are 1's output 1, else 0

Analysis:
Since $\vec{r}_i$'s are uniform in $\mathbb{Z}_q^n$ then in every run the procedure D sees an oracle which is either the Uniform (if $\mathcal{O}$ is) or $LWE_{\vec{s}}$ for independent random $\vec{s}$ (if $\mathcal{O} = LWE_{\vec{s}}$). Hence when $\mathcal{O}$ = Uniform the in each run it outputs 1 w.p. $\leq \tau - \varepsilon/4$, and when $\mathcal{O} = LWE_{\vec{s}}$ it outputs 1 w.p. $\geq \tau + \varepsilon/4$ in each run. By Chernoff, after $poly(n, 1/\varepsilon)$ runs the average is closer to the right answer than $\varepsilon/4$, except with probability $\exp(-n)$. ⊠

Thm 2: When $q = O(poly(n))$, given a distinguisher for $D\text{-}LWE[n,q,\alpha]$ with advantage $1-negl(n)$, can construct a solver for $S\text{-}LWE[n,q,\alpha]$ with success $1-negl(n)$, paying a factor nq in running time.

Proof idea:
Can check if $s_i = k$ for any $i \in \{1,\dots,n\}$, $k \in \mathbb{Z}_q$ using the transformation

$$\varphi_{r,i}(\vec{a},b) = (\vec{a} + (\underbrace{0 \cdots 0\, r\, 0 \cdots 0}_{r \text{ in } i\text{-th position}}),\ b + rk) \quad (\text{everything mod } q)$$

When $(\vec{a},b) \in_R LWE_{\vec{s}}$ then the same holds for $\Psi_{r,i}(\vec{a},b)$ if $s_i = k$ but if $s_i \neq k$ and r is random then $\Psi_{r,i}(\vec{a},b) \in_R$ Uniform. ⊗

Thm 3 [Peikert 2009]: When $q = q_1 q_2 \cdots q_\ell$ such that the q_i's are distinct primes, for each i $\frac{1}{\alpha} \cdot \sqrt{\log n} \ll q_i \leq B = poly(n)$, then given a distinguisher D for $D\text{-}LWE[n, \alpha, q]$ with advantage $1 - negl(n)$, can construct a solver S for $S\text{-}LWE[n, \alpha, q]$ with success prob. $1 - negl(n)$, paying a factor $poly(n, B, \log q)$ in running time.

Proof:

Note that we can factor q in time $poly(B, \log q)$. We begin by describing a subroutine $Test(i, j)$ that uses the distinguisher D to decide if $s_i = 0 \pmod{q_j}$.

$Test(i, j)$ has access to $LWE_{\vec{s}}$ oracle and the distinguisher D.

1. Run D, answering its O-queries as follows:
   (i) Choose $r \in_R \mathbb{Z}_{q_j}$ and use your oracle to get $(\vec{a}, b) \in_R LWE_{\vec{s}}$
   (ii) Set $\vec{a'} = \vec{a} + \underbrace{(0 \cdots r \cdots 0)}_{r \text{ in position } i} \cdot q/q_j \pmod q$, return $(\vec{a'}, b)$ to D
2. Output whatever D does

Analysis of $Test(i, j)$: Note that $\vec{a'}$ is uniform in $\mathbb{Z}_q^n$.

* If $s_i = 0 \pmod{q_j}$ then $\langle \vec{a'}, \vec{s} \rangle = \langle \vec{a}, \vec{s} \rangle \bmod q$, hence $(\vec{a'}, b) \in_R LWE_{\vec{s}}$
* If $s_i \neq 0 \pmod{q_j}$ then conditioned on any fixed value of $\vec{a'}$ we have:

$$b = \underbrace{\langle \vec{a'}, \vec{s} \rangle}_{\text{fixed value}} - \underbrace{(s_i \cdot r) \cdot q/q_j + e}_{\text{"error" term}} \pmod q$$

Since q_j is prime and $r \in_R \mathbb{Z}_{q_j}$ then $s_i \cdot r$ is uniform in $\mathbb{Z}_{q_j}$, hence $(s_i \cdot r) q/q_j + e$ is obtained by choosing at random a point x in the set $\{0, q/q_j, 2q/q_j, \ldots, \frac{(q_j - 1)q}{q_j}\}$, then adding to it $e \in_R \Phi_{\alpha q}$ to get $x + e \pmod q$.

Note that the condition $q_j \gg \frac{1}{\alpha} \cdot \sqrt{\log n}$ means that the variance of e is much larger than $(q/q_j)^2$, hence $(e \bmod q/q_j)$ is nearly uniform and so the error term is nearly uniform in $\mathbb{Z}_q$

Φ must be a smooth distribution

$\boxtimes$

Now that we have a reliable $Test(i,j)$, we can use it to compute $(s_i \bmod q_j)$ for all $i \in \{1,2,\cdots,n\}$ and $j \in \{1 \cdots l\}$, and then recover $\vec{s}$ using Chinese Remaindering. To compute $s_i \bmod q_j$ we use the subroutine $S(i,j)$ as follows:

$S(i,j)$ has access to $LWE_{\vec{s}}$ oracle and the procedure $Test(i,j)$

1. For $K=0$ to $q_j - 1$ do (check if $s_i = K \bmod q_j$)
   (i) Run $Test(i,j)$, answering its LWE-oracle queries as follows:
      (a) Use your oracle to get $(\vec{a}, b) \in_R LWE_{\vec{s}}$
      (b) Set $b' = b + \langle \vec{a}, (0 \cdots 0, -K \cdots 0) \rangle \pmod q$, reply with $(\vec{a}, b')$
   (ii) If $Test(i,j)$ says "$s_i = B \bmod q_j$" output K and halt
   else proceed to next value of K

Analysis of $S(i,j)$: The $Test(i,j)$ subroutine sees samples from $LWE_{\vec{s}-\vec{t}}$, where $\vec{t} = (0, \cdots 0, K \cdots 0)$ ☒

Hardness of LWE: The main reduction is from [Regev 2005]: Given LWE-solver and oracle for sampling random short points in Λ^*, can solve Bounded Distance Decoding (BDD) in Λ. The reduction relies on some properties of the error distribution of LWE.

* The LWE error distribution $\Phi_{\alpha q}$ is a projection of a spherical distribution $D_{\alpha q}$ (in n-dimensions) onto the first coordinate.
* $D_{\alpha q}$ is "smooth" in the following sense:
  - Let Λ be an arbitrary lattice (or coset of lattice) with $\lambda_n(\Lambda) \ll \alpha q$
  - Denote by $\tilde{D}_{\Lambda,r}$ the discrete version of D_r, i.e. D_r conditioned on Λ.
  - Then if we choose $\vec{x} \in_R \tilde{D}_{\Lambda,r}$ and $\vec{y} \in D_s$, such that $r^2+s^2=(\alpha q)^2$ the induced distribution on $\vec{x}+\vec{y}$ is close to $D_{\alpha q}$.
  - Also, $D_{\alpha q}$ mod B is nearly uniform for any basis B
* The n-dimensional Gaussian has this property, where "$\lambda_n \ll \alpha q$" means $\lambda_n \cdot \omega(\sqrt{\log n}) < \alpha q$. In this case $\Phi_{\alpha q}$ is just the (1-dim) continuous Gaussian with variance αq.
* The uniform distribution over a sphere of radius $\sqrt{n} \cdot \alpha q$ also has this property, for worse parameters ($\lambda_n \cdot \text{super-poly}(n) < \alpha q$).

Lemma 1: Let n, α, q be parameters as before. There is an efficient algorithm that given as input a basis B for n-dimension lattice $\Lambda = \Lambda(B)$, another parameter $r \gg q/\lambda_1(\Lambda)$, a point $\vec{x} \in \mathbb{R}^n$ s.t. $\text{dist}(\vec{x}, \Lambda) \leq \frac{\alpha q}{\sqrt{2}r}$, and access to two oracles:
- A "global" solver for LWE$[n, \alpha, q]$ (independent of Λ)
- A "lattice specific" oracle that samples from $\tilde{D}_{\Lambda^*,r}$

outputs the (unique) $\vec{v} \in \Lambda$ closest to $\vec{x}$ with probability $1-\text{negl}(n)$.

We will sketch the proof later.

* The oracle for sampling from $\tilde{D}_{\Lambda,r}$ can be implemented, e.g., using a procedure from [Gentry-Peikert-Vaikuntanathan-2008]

Theorem 4 [GPV 2008]: Given a basis B for a lattice $\Lambda = \Lambda(B)$, we can efficiently sample from the discrete Gaussian distribution $D_{\Lambda,r}$ (upto negl(n) statistical distance) for any parameter $r \gg \max_i \|b_i\|$.

* For Gaussians, it is enough to have $r \geq \max_i \|b_i\| \cdot \omega(\sqrt{\log n})$. For Uniform in a sphere we can get the same result but worse parameters.

* Using LLL we can always find a basis B^* for Λ^* such that $\max_i \|b_i^*\| \leq 2^{n/2} \cdot \lambda_1(\Lambda^*) \leq 2^{n/2} \cdot n/\lambda_1(\Lambda)$, so we can use the GPV sampler to implement the oracle for $\tilde{D}_{\Lambda,r}$ whenever $r \geq 2^n/\lambda_1(\Lambda)$.

Thm 5 [Peikert 2008]: Let $\alpha = 1/\text{poly}(n)$, $\gamma = n/\alpha$, $q = \exp(n)$. Then given access to a solver S for $LWE[n, \alpha, q]$ and a basis B for an arbitrary lattice $\Lambda = \Lambda(B)$ (in n dimensions), we can approximate $\lambda_1(\Lambda)$ to within a factor of γ.

* Important comment: The procedure approximates the number $\lambda_1(\Lambda)$, it does not find a vector in Λ of that length.

Proof: First use LLL to get a $2^{n/2}$-approximation for $\lambda_1(\Lambda)$, namely a number b_0 s.t. $\lambda_1 \leq b_0 < 2^{n/2}\lambda_1$. Then define $b_i = b_0/\gamma^i$. We next describe a procedure that distinguishes the cases $\lambda_1 < b_{i+1}$ from $\lambda_1 \geq b_i$. Run this procedure with $i = 0, 1, 2, \ldots$, let i^* be the first index for which the procedure says "$\lambda_1 > b_{i^*}$", output b_{i^*}. Clearly if $\lambda_1 \in [b_{i+1}, b_i)$ then this procedure will output either b_{i+1} or b_i.

* A procedure for distinguishing $\lambda_1 < d$ from $\lambda_1 > d \cdot \gamma$:

Distinguish(B, d): Using a solver for LWE[n, α, q]

1. For $i = 1$ to $N = \text{poly}(n)$ do
   (i) Set $d' = d \cdot \sqrt{n}/2$, choose $\vec{w}_i \in_R$ Sphere(radius-d') $\subseteq \mathbb{R}^n$
   (ii) Set $\vec{x} = \vec{w} \bmod B$
   (iii) Run the algorithm from Lemma 1 on input basis B, parameter $r = \frac{q \cdot \sqrt{2n}}{d \cdot \gamma}$, point $\vec{x}$, and the two oracles:
   - LWE solver, the one that you have access to
   - The GPV sampler, using LLL-reduced basis for B^*
   (iv) Let $\vec{v}$ be the point that the algorithm returns
   Set $\text{vote}_i = "\lambda_1 \geq d\gamma"$ if $\vec{x} - \vec{v} = \pm \vec{w}$, else $\text{vote}_i = "\lambda_1 < d"$
2. If all votes say "$\lambda_1 \geq d \cdot \gamma$" then output "$\lambda_1 \geq d \cdot \gamma$", else "$\lambda_1 < d$".

Analysis: We show that (a) when $\lambda_1 \geq d \cdot \gamma$ then all the conditions of Lemma 1 are satisfied and $\vec{x} \pm \vec{w}$ is the closest lattice point to $\vec{x}$, so the reduction will always return $\vec{x} \pm \vec{w}$, and (b) when $\lambda_1 < d$ then the view of the algorithm from Lemma 1 does not determine a unique $\vec{w}$, so with non-negligible probability it returns a different point $\vec{v}$.

Case (a): $\lambda_1 \geq d \cdot \gamma$:

Recall that $\text{dist}(\vec{x}, \Lambda) < d\sqrt{n}/2 = d \cdot \frac{2 \cdot \gamma \cdot q}{\sqrt{n} \cdot q \cdot \sqrt{2n}} = \frac{2q}{\sqrt{2}\, r}$, as needed for the lemma. Also $r = \frac{q\sqrt{2n}}{\gamma \cdot d} > \frac{q\sqrt{2n}}{\lambda_1}$. Finally, since $q = \exp(n)$ then the GPV-sampler gives good enough samples. Hence the reduction works and we always get the unique closest point to $\vec{x}$, which is $\vec{x} \pm \vec{w}$.

Case(b): $\lambda_1 < d$: Let $\vec{y}$ be the shortest nonzero point in Λ, $\|\vec{y}\| < d$, then with non-negligible probability both $\vec{x} - \vec{w}$ and $\vec{x} - \vec{y} - \vec{w}$ are within d' away from $\vec{x}$, hence both equally likely given $\vec{x}$ ⊠

Proof sketch for Lemma 1: Let $\vec{v} \in \Lambda$ be the point closest to $\vec{x}$, let $\vec{t}$ be the coefficient vector of $\vec{v}$ base B (i.e., $\vec{v} = B\vec{t}$) and denote $\vec{s} = \vec{t} \bmod q$. We next show a subroutine that uses the sampler from $\tilde{D}_{\Lambda^*,r}$ to generate instances from $LWE_{\vec{s}}$. Then we can use the LWE-solver to find $\vec{s}$. (For $q = \exp(n)$ we can show that $\vec{s} = \vec{t}$, otherwise there is still work to do.)

LWE-Generate$(B, \vec{x})$: having access to $\tilde{D}_{\Lambda^*,r}$ sampler

1. Draw a sample $\vec{y} \in_R \tilde{D}_{\Lambda^*,r}$, let $\vec{a}$ be the coefficients of $\vec{y}$ to the base B^*, reduced mod q, $\vec{a} = B^t \vec{y} \pmod q$
2. Draw an error term $e \in_R \Phi_{\alpha/2\sqrt{\pi}}$
3. Output $(\vec{a}, b = \langle \vec{x}, \vec{y} \rangle + e \pmod q)$

Claim: The output of LWE-Generate is statistically close to $LWE_{\vec{s}}$, except that the error parameter could be some $\beta \le \alpha$.

Proof: We need to show that $\vec{a}$ is nearly uniform in $\mathbb{Z}_q^n$, and that conditioned on $\vec{a}$, the value of b is distributed almost according to $\langle \vec{a}, \vec{s} \rangle + \Phi_\beta \pmod q$ for some $\beta \le \alpha$.

(A) Consider the lattice $q \cdot \Lambda^*$ and all its q^n cosets

$$\vec{a}\text{-coset} = \{ B^*\vec{a} + q \cdot \Lambda^* \} = \{ B^*\vec{z} \mid \vec{z} \in \mathbb{Z}^n, \vec{z} = \vec{a} \pmod q \} \overset{\det}{=} \Lambda^* + \vec{a}$$

Then the $\vec{a}$ output of LWE-Generate is exactly the coset of $\vec{y}$. Since the parameter r is large $r \gg q/\lambda_1(\Lambda) \ge \frac{q\,\lambda_n(\Lambda^*)}{n}$ then $\tilde{D}_{\Lambda^*,r}$ is nearly uniform among the cosets, hence $\vec{a}$ is nearly uniform.

(B) Conditioned on any fixed $\vec{a}$, the vector $\vec{y}$ is chosen from the discrete distribution on the $\vec{a}$-coset, $\tilde{D}_{\Lambda^*+\vec{a},r}$. Denoting $\vec{x} = \vec{v} + \vec{w}$ we have

$$\langle \vec{x}, \vec{y} \rangle = \langle \vec{v} + \vec{w}, \vec{y} \rangle = \langle \vec{v}, \vec{y} \rangle + \langle \vec{w}, \vec{y} \rangle$$
$$= \langle \vec{s}, \vec{a} \rangle + \langle \vec{w}, \vec{y} \rangle \pmod q$$

hence $b = \langle \vec{s}, \vec{a} \rangle + \langle \vec{w}, \vec{y} \rangle + e \pmod q$

("fixed" / "random")

Recall that $\Phi_{\alpha/2\sqrt{n}}$ is the projection of $\mathcal{D}_{\alpha/2\sqrt{n}}$ onto the 1'st coordinate, namely $\langle \vec{e}_1, \mathcal{D}_{\alpha/2\sqrt{n}} \rangle$, and since $\mathcal{D}$ is spherical then this is also the same as $\langle \vec{u}, \mathcal{D}_{\alpha/2\sqrt{n}} \rangle$ for any other unit vector $\vec{u}$. In particular, $\Phi_{\alpha/2\sqrt{n}} \equiv \langle \vec{w}, \mathcal{D}_{\alpha/2\sqrt{n}} \rangle \cdot \frac{1}{\|\vec{w}\|} \equiv \langle \vec{w}, \mathcal{D}_{\frac{\alpha}{2\sqrt{n}\cdot\|\vec{w}\|}} \rangle$

Hence $\langle \vec{w}, \vec{y} \rangle + e \equiv \langle \vec{w}, \vec{y} \rangle + \langle \vec{w}, \vec{z} \rangle = \langle \vec{w}, \vec{y}+\vec{z} \rangle$ where $y \in_R \tilde{\mathcal{D}}_{q\Lambda^*+\vec{a}, r}$ and $z \in_R \mathcal{D}_s$ where $s = \frac{\alpha}{2\sqrt{n}\|\vec{w}\|}$. Now $\|\vec{w}\|$ is "short" so s is "large". The parameters chosen so r, s are large enough so that $\tilde{\mathcal{D}}_{q\Lambda^*+\vec{a}, r} + \mathcal{D}_s$ is close to the continuous $\mathcal{D}_t$ where $t = \sqrt{r^2+s^2}$.

Therefore $\langle \vec{w}, \vec{y} \rangle + e \approx \langle \vec{w}, \mathcal{D}_t \rangle = \Phi_{\|\vec{w}\|\cdot t}$, and again the parameters are such that $\|\vec{w}\| \cdot t \leq \alpha$. ⊠